Abstract

Term rewriting is a computation model for equational reasoning. This model underlies various fields in computer science, including functional programming, automated theorem proving, and software verification based on equational specifications. In rewriting there are two fundamental properties. One is termination, which ensures finiteness of computation steps. The other is confluence. It guarantees uniqueness of computation results without relying on any specific computational strategy. For a computational system with non-determinism, confluence corresponds to consistency of the system. It plays a key role as well-definedness of function definitions and correctness of specifications.

In this thesis we present a new approach for analyzing confluence of left-linear term rewrite systems based on compositional confluence criteria. A compositional confluence criterion means a sufficient condition that, given a rewrite system and its subsystem, confluence of the subsystem implies confluence of the original system. Since such a subsystem can be analyzed by any other (compositional) confluence criterion, compositional confluence criteria can be applied to subsystems successively. This method enables us to decompose a rewrite system into its subsystem for showing confluence of the original one.

In order to obtain compositional confluence criteria, we develop a variant of decreasing diagrams method. It is known that most of confluence criteria for left-linear rewrite systems can be shown by the method. Exploiting this fact, we demonstrate how those confluence criteria can be recast into compositional criteria by adopting a compositional version of the decreasing diagram method in their proofs. Furthermore we show how existing confluence criteria based on decreasing diagrams are generalized to ones composable with other criteria. We also show how such a criterion can be used as a reduction method to remove rewrite rules unnecessary for confluence analysis. Effectiveness of these approaches is assessed by experimental data based on our confluence tool Hakusan. In addition to these contributions, we prove that Toyama's parallel closedness result based on parallel critical pairs subsumes his almost parallel closedness theorem.

keywords: *term rewriting, confluence, decreasing diagrams, parallel closedness, automation.*