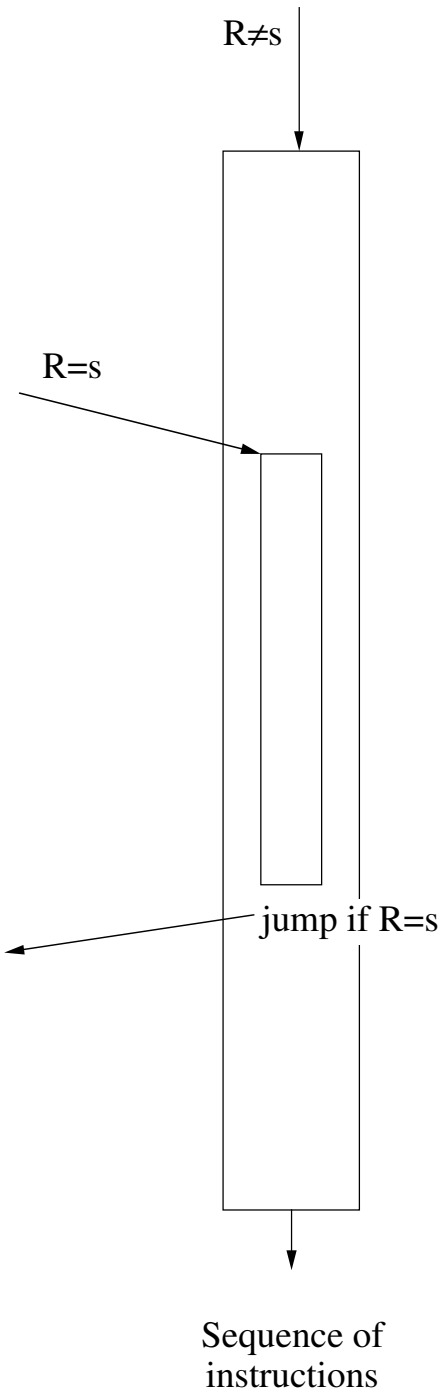
How to compile unification: The two-stream idea



- Given a sequence of instructions.
- It is possible to efficiently execute any contiguous subsequence. Give each subsequence a unique identifier *s*. Then a single comparison per subsequence and a single register R are all that is needed.
- Arrange the unification instructions in a depth-first traversal of the term. Then two properties are true:
  - Each subterm is a contiguous sequence of instructions.
  - Nested subterms are also nested sequences of instructions. (This permits further reduction of overhead.)
- Idea due to Mohamed Amraoui, André Mariën and Bart Demoen, Kent Boortz, and Micha Meier.