

Automatic Code Generation for Graph Algorithms

Research Problem

Turning mathematical graph algorithms into actual implementations that run at speed is complicated. It requires:

1. **algorithmic design** to identify the appropriate implementable algorithms
2. **tuned implementations** that consider data storage formats and available hardware features

Target Problem

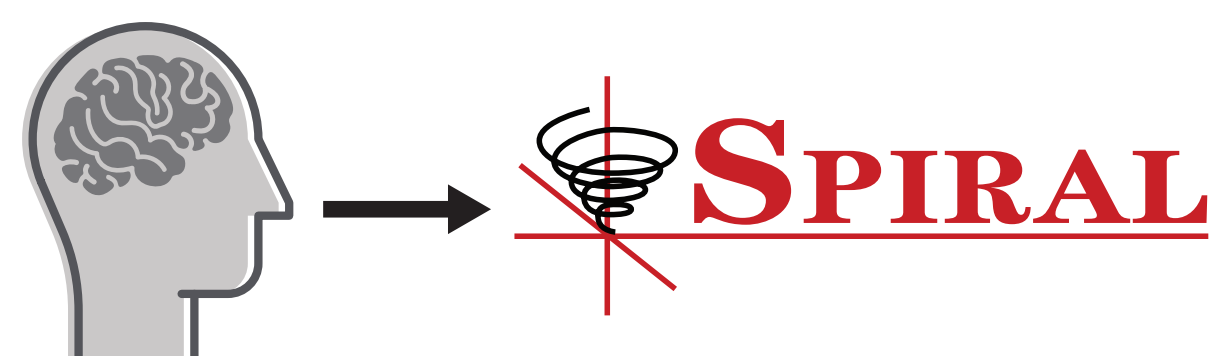
Using triangle counting as an example, we demonstrate our approach to generating graph algorithms from their mathematical specification.

Mathematical Specification

$$\Delta = \frac{1}{6} \Gamma(A^3)$$

Proposed Solution

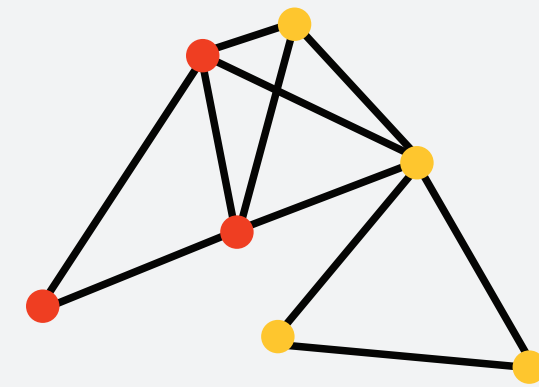
Encode expert knowledge about algorithm design and optimization into an automated system (SPIRAL) to generate tuned implementations automatically. Allow the use of GraphBLAS formulae for providing mathematical specifications.



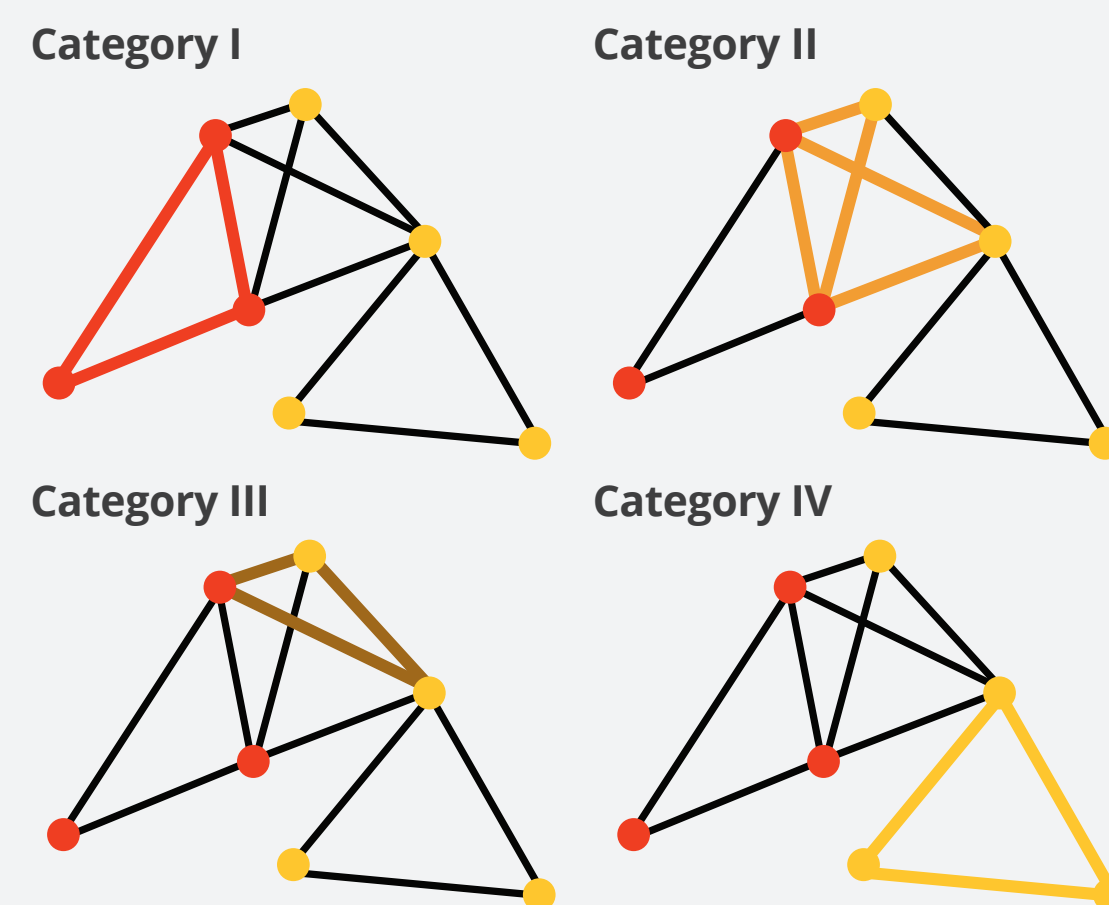
Algorithmic Design

Formally deriving algorithms from the mathematical specification.

Example: Triangle Counting



Original graph split into two subgraphs: processed (red) and unprocessed (yellow)

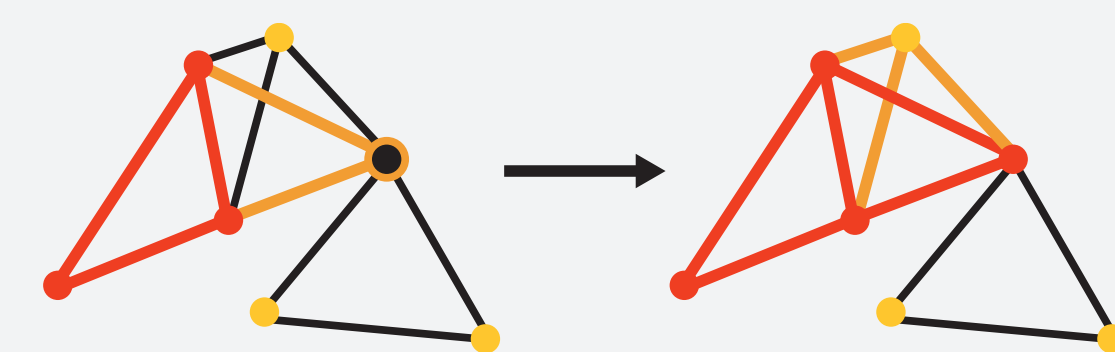


Intuition

Counting different categories of triangles as we iterate over the different vertices yields different algorithms.

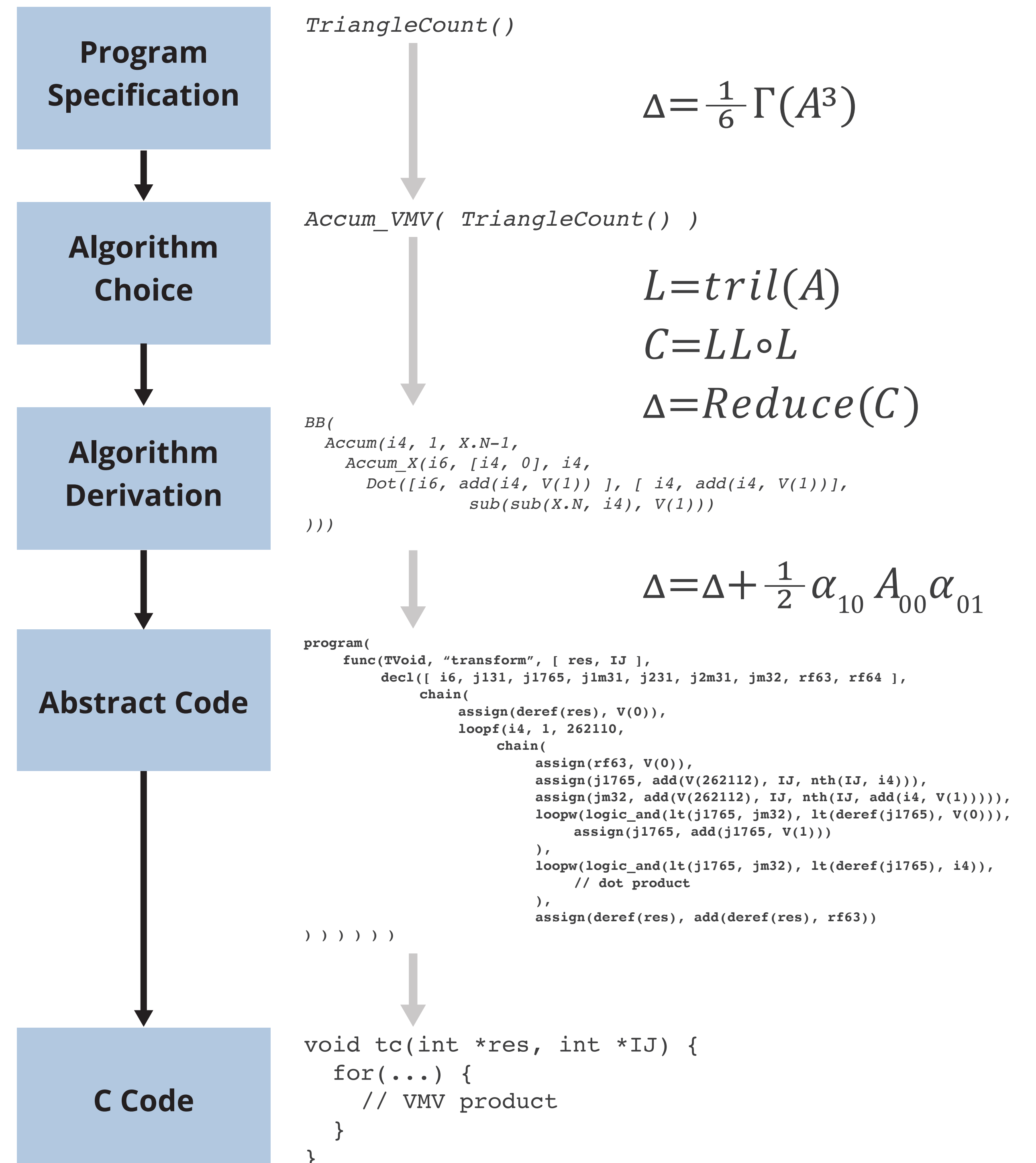
Illustration

Counting Category I and II triangles, where red vertices have been processed.



Automatic Code Generation

Formalize the algorithm and implementation techniques into SPIRAL



Copyright 2018 Carnegie Mellon University. All Rights Reserved.

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

The view, opinions, and/or findings contained in this material are those of the author(s) and should not be construed as an official Government position, policy, or decision, unless designated by other documentation.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

Internal use:* Permission to reproduce this material and to prepare derivative works from this material for internal use is granted, provided the copyright and "No Warranty" statements are included with all reproductions and derivative works.

External use:* This material may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other external and/or commercial use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

* These restrictions do not apply to U.S. government entities.

Carnegie Mellon® is registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.
DM18-1129