

# Implementing K-Trusses

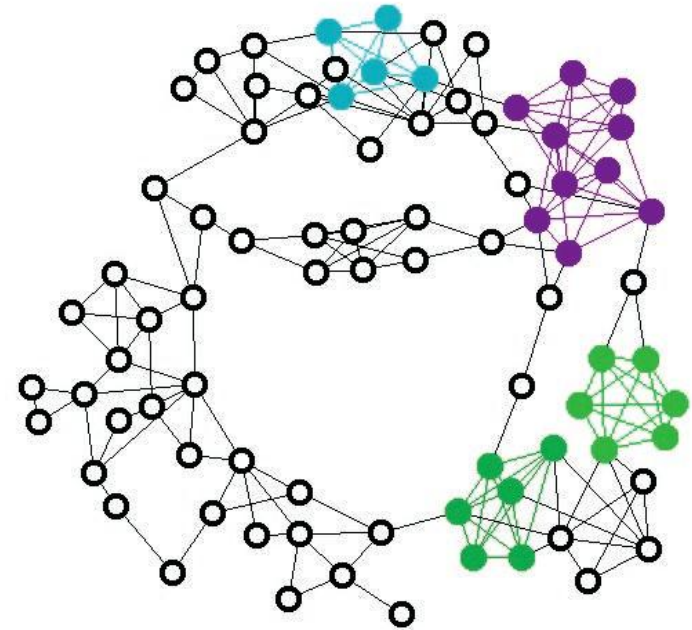
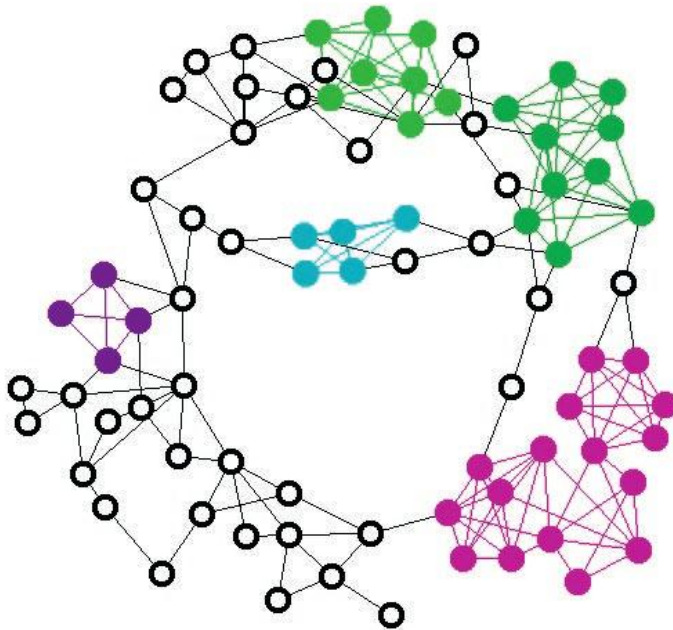
Tillmann Fiehn  
Sebastian Arnold



Fachgebiet Datenbanksysteme und Informationsmanagement  
Technische Universität Berlin

<http://www.dima.tu-berlin.de/>

- Relaxation of k-member clique
- Non-trivial, single-component maximal subgraph
- Every edge is contained
  - in at least  $k-2$  triangles
  - in the subgraph



- findKTrusses: Edges,  $k \rightarrow$  ZoneAssignments ( $v, z$ )
  - finds all  $k$ -trusses of the graph
  - each returned vertex  $v$  is part of a truss  $z$

simplifyGraph

while true do:

    augmentGraphWithDegrees

    enumerateTriangles

    keep only edges contained in  $k-2$  triangles

    if all edges kept break;

done

findComponents

init

## SimplifyGraphJob



remove loops

drop duplicate  
edges



while maybe unsupported edges

## CalculateDegreesJob



send to each  
vertex

sum degrees



join degrees



## EnumerateTrianglesJob



send to  
lower vertex

build  
open triads



join triads  
and edges



## FindKTrussesJob



split triangles  
to edges



drop  
unsupported edges



if no edges dropped **break**

while maybe zones connected

## FindComponentsJob



identity of zones  
send edge to each vertex



assign one zone  
to edges



find interzone  
edges



if no zones connected **break**



bin vertices and  
zone assignments

assign best zone  
to each vertex

