Hypergraph Mining

D.Papadimitriou
(dimitri.papadimitriou@alcatel-lucent.com)

Graph-based modeling

- Graph-based modeling provides
  - Foundation for phenomena and/or problems involving one-to-one relationships (functional) and/or interactions (dynamic) among entities
  - Allows data analysis and mining to understand relations between these entities -> Graph mining
- In communication networks, "dyadic" deterministic graphs but other types of graphs exist (e.g. Cayley graph, stochastic graphs, bipartite graphs, etc.)

Graphs

- **Unweighted Graph** $G = (V,E)$
  - V : set of vertices, $|V| = n$
  - E : set of edges, $|E| = m$
    - Elements of E are pairs (u,v) where $u,v \in V$
    - An edge (v,v) is a self-loop
- **Weighted Graph** $G = (V,E,\omega)$
  - V : set of vertices, $|V| = n$, E : set of edges, $|E| = m$
  - ω = function which associates to each edge a weight

- **Undirected graph**
  - The edge pairs are unordered
    - E defines symmetric relation
    - $(u,v) \in E$ implies $(v,u) \in E$, (u,v) and (v,u) corr. to the same edge
- **Directed graph** (digraph)
  - The edge pairs are ordered

Example: network modeling

- Network topology modeled as undirected unweighted graph G = (V,E)
  - **AS-level topology**: vertices (abstract nodes) set V, |V| = n, represents the autonomous systems (AS), and edges (or links) set E, |E| = m, represents the interconnection between AS pairs (u,v), u, v $\in$ V

- Network topology modeled as undirected weighted graph G = (V,E,ω)
  - **Router-level topology**: vertices (nodes) set V, |V| = n, represents routers or inter-connection points, and edges (or links) set E, |E| = m, represents nodes interconnection

Example: path modeling

- **Path** from source s to destination t, $p(v_0=s,v_m=t)$: node sequence $[v_0(=s),v_1,\ldots,v_{i-1}=u,v_i,\ldots,v_m(=t)]$ such that v_i is adjacent to v_{i-1}, $(v_{i-1},v_i) \in E(G)$, $\forall$ i

- Distinction between topological path and routing path (output of the routing algorithm)

  -> routing topology is a sub-graph of the graph representing the network topology

- **Diameter Δ(G)**: maximum length of the shortest (topological) path p(u,v) between any two pair of vertices (u,v), u, v $\in$ V

Limits of (Dyadic) Graph Modeling

- Graph-based modeling fails to capture group-level interactions / relationships between entities that are of different nature

- Many of the relationships exhibited are not restricted to be one-to-one, in particular in communication networks
  - multi-layer structures
  - multi-level/hierarchical structures
  - (hidden) relationships between entities

Objective

- Build a model that inherently handles many-to-many relationships/group interactions -> hypergraphs
- In a graph an edge can be incident on exactly two vertices whereas each hyperedge in a hypergraph is an arbitrary subset of the vertex set and represents relations between its elements
- Many hyperedges may be subsets of other hyperedges

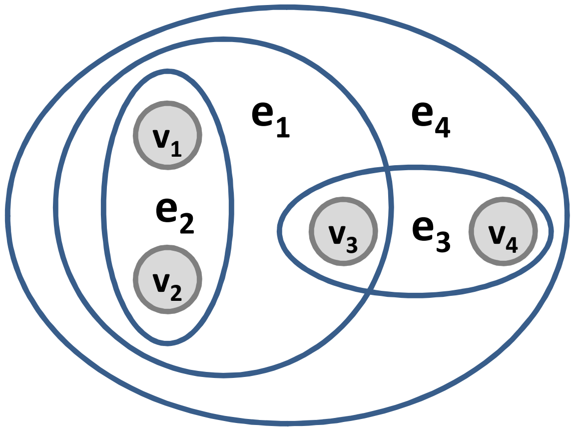
- Hypergraphs can model many-to-many relationships among entities enabling in turn to handling problems such as
  - Similarity
  - Clustering
  - Construction of classifiers

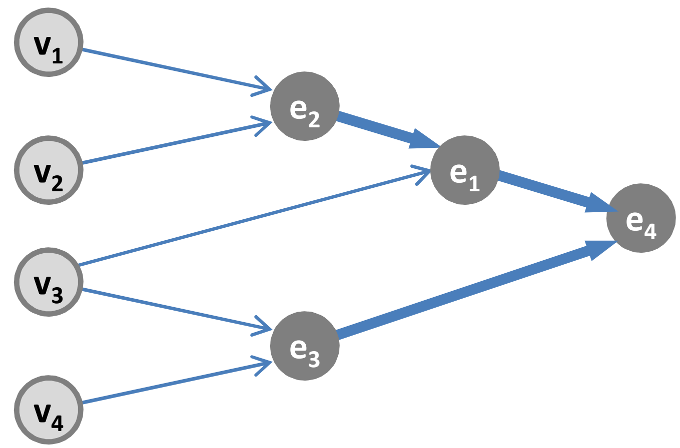
Hypergraph definition

- V : finite set of vertices
- E : family of subsets of V such that $U_{e \in E}$ = (V,E,ω) is called a hypergraph with hyperedge set E
  - When each hyperedge $e \in E$ is assigned a positive weight $\omega(e)$, weighted hypergraph
- Notation:
  - Hypergraph H = (V,E)
  - Weighted hypergraph H = (V,E,ω)
- A hypergraph can be represented by a $|V| \times |E|$ incidence matrix H_t:
  - $h_t(v_i,e_j) = 1$, if $v_i \in e_j$
  - $h_t(v_i,e_j) = 0$, if $v_i \notin e_j$

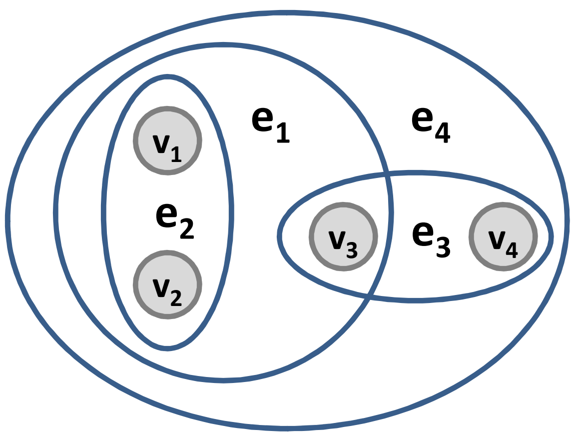
Other representations

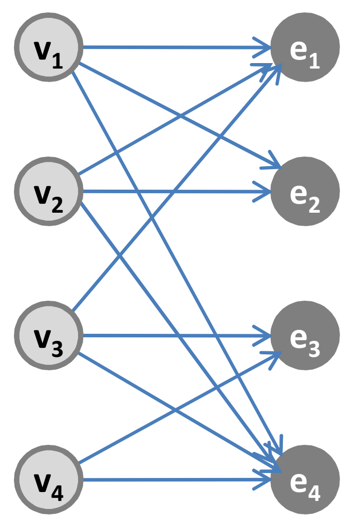
- Hierarchical DAG (Directed acyclic graph)





- Bipartite





See also: Beyond Graphs: Toward Scalable Hypergraph Analysis, B.Heintz and A.Chandra Systems

Shared Risk Model: Groups

- Let denote by
  - C : set of components of the system, $C = \{c_1,\ldots,c_p\}$ such that $|C| = p$
  - S : set of shared risk groups, $S = \{s_1,\ldots,s_q\}$ such that $|S| = q$

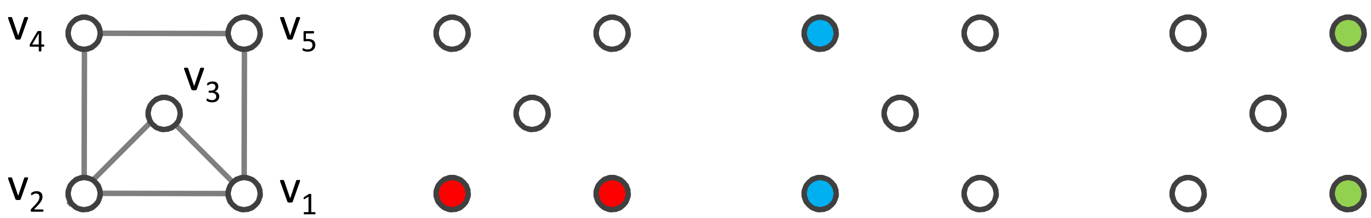
- Element $c_j \in C$ belongs to SRG s_i if c_j includes resources/supplies covered by s_i

- Properties
  - Any component $c_i \in C$ belongs at least to one SRG, i.e., $|S| = q \geq p$
  - By extension, $c_i \in C$ belongs to SRG set $s' = \{s_1,\ldots,s_{q'}\}|_{q' \leq q}$ if c_i crosses at least one of the resources of each of its members $s_1,\ldots,s_{q'}$
  - Any pair of elements c_i, $c_j \in C$ belonging to the SRG s_k ($\{c_i, c_j\} \in s_k$) can individually belong to a set of other SRGs, i.e., $c_i \in s_p$, $c_j \in s_q$ such that $s_k \cap s_p = \{c_i\}$ and $s_k \cap s_q = \{c_j\}$
  - More generally any component from a given subset of components taken individually may belong to other SRGs

Shared risk models

- SRG: multiple "entities" sharing common risk

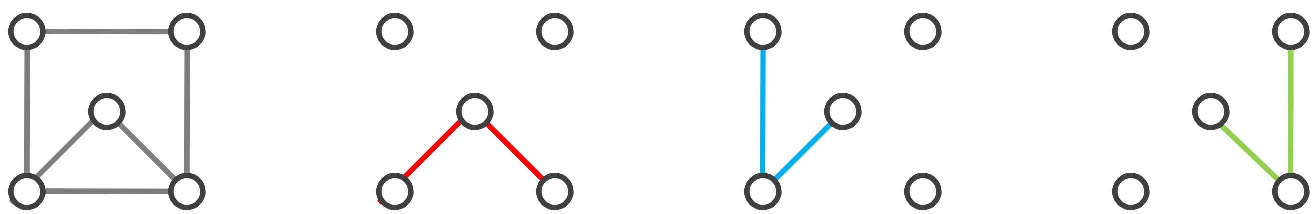
i) Nodal



$s_1 = \{v_1,v_2\}$ $s_2 = \{v_2,v_4\}$ $s_3 = \{v_1,v_5\}$

Components $C = \{v_1,v_2,v_3,v_4,v_5\}$

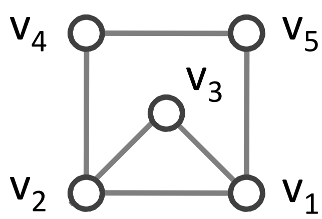
ii) Link

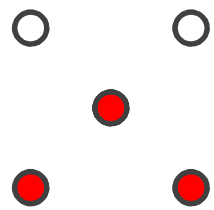


Shared risk models: nodal

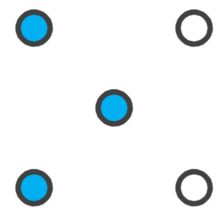
- Application is "software failures" (programmable nodes)

Nodal

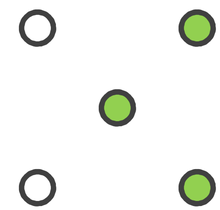




$s_1 = \{v_1,v_2,v_3\}$



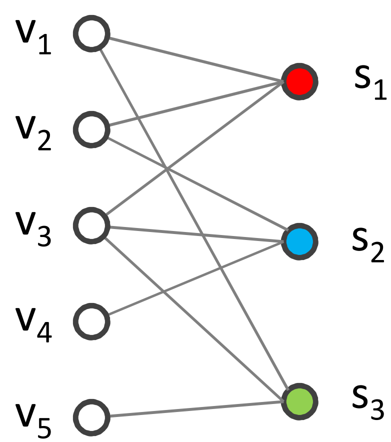
$s_2 = \{v_2,v_3,v_4\}$



$s_3 = \{v_1,v_3,v_5\}$

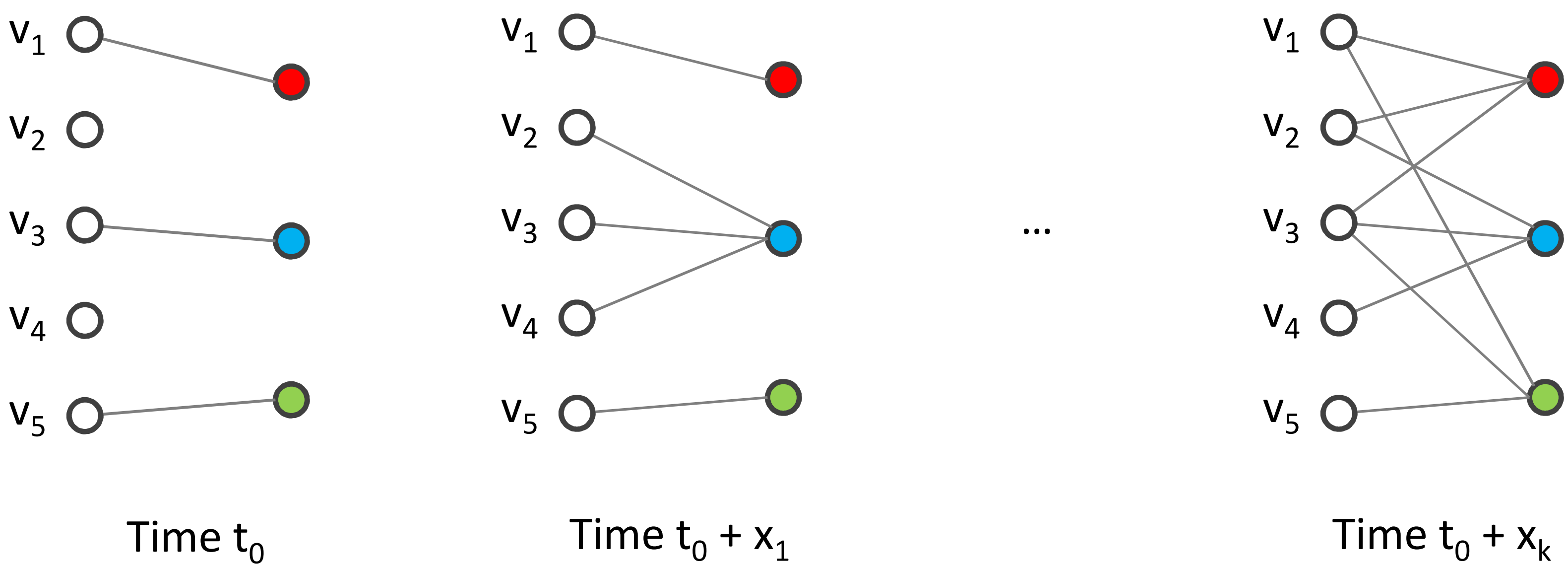
Bipartite representation

- Components $C = \{v_1,v_2,v_3,v_4,v_5\} \equiv$ vertices of the hypergraph
- SRG $S = \{s_1,s_2,s_3\} \equiv$ Hyperedges of the hypergraph $e_1 \equiv s_{1,}\ e_2 \equiv s_{1,}\ e_2 \equiv S_3$



Procedure

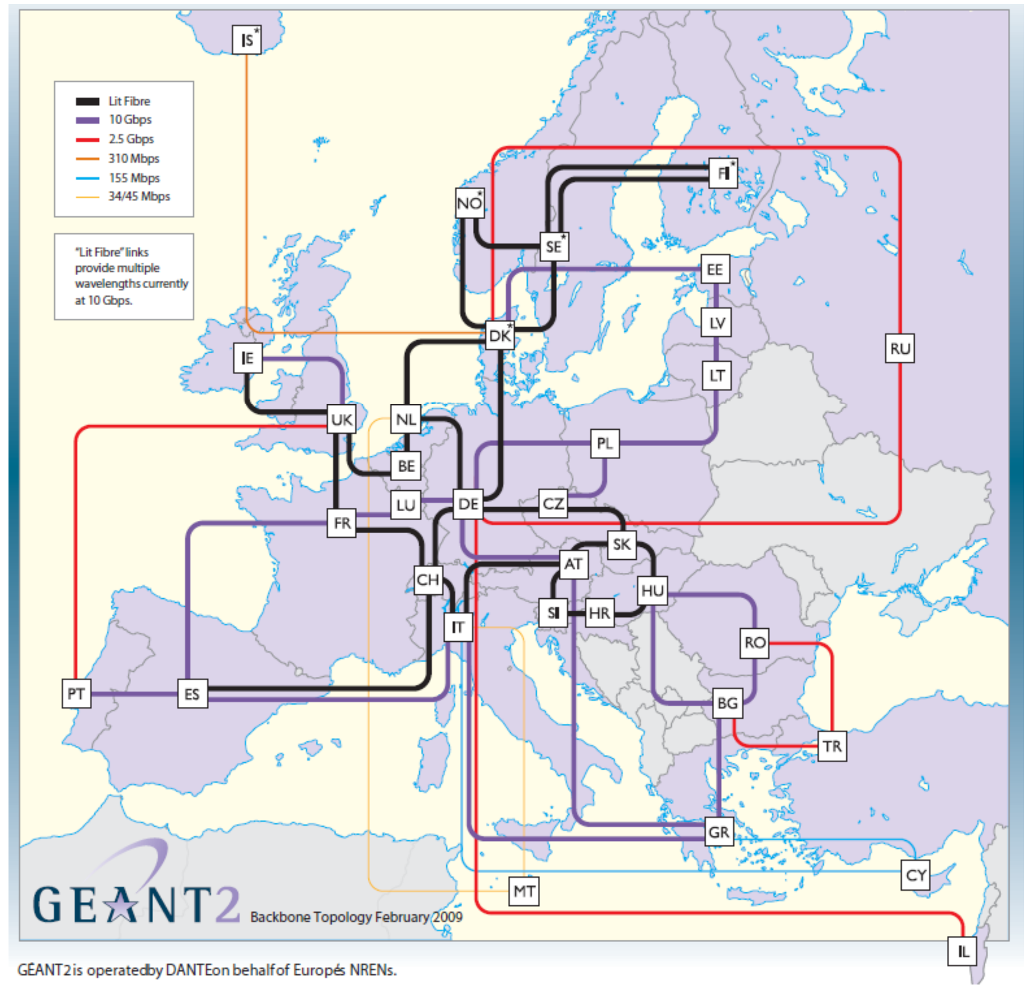
- Iterative construction (joint failure events)



- Note: single "failure" can also occur

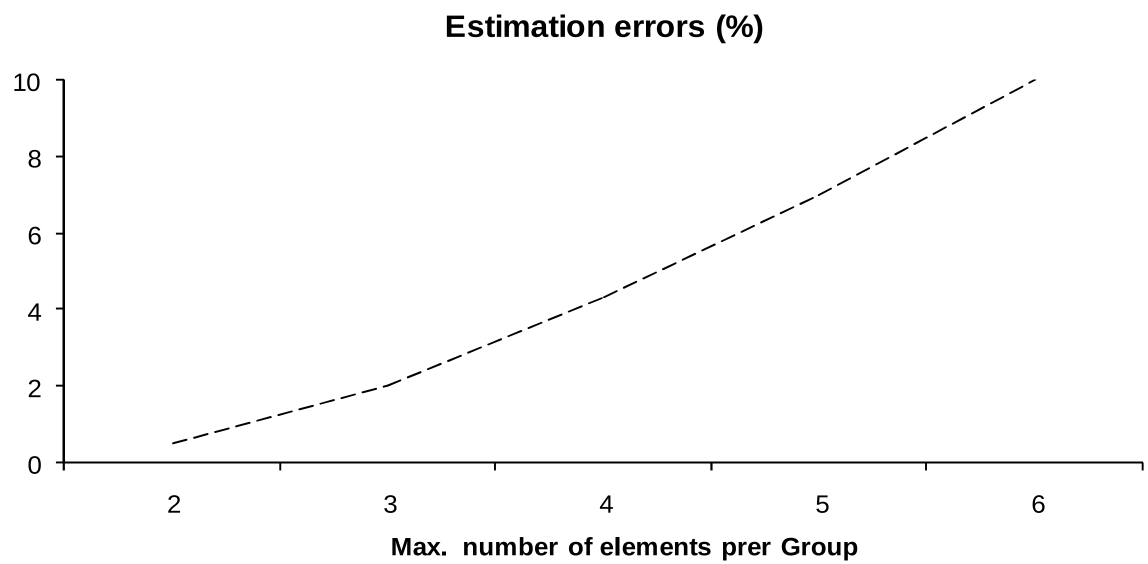
Setup

- Setup based on GEANT2 network topology (comprising 32 physical nodes)
- Shared risk groups comprising up to 6 shared components (i.e. a node can include up to 6 components common to other nodes)
- If that component fails on a given node, it could also fail on the others (if sharing common root cause)



Results

- Estimation error vs number of shared components per group (from 2 to 6)



  - Relatively good detection accuracy of joint failure events for groups of 2 and 3 components with ν parameter set to 2 (higher value of this parameter does not further increase accuracy)
  - Prediction error increases as the number of components per group increases (about 10% for p=6)

Limits of Deterministic Hypergraphs

- Conventional hypergraph structure assigns vertex v_i to hyperedge e_j with a **binary decision**, i.e., $h_t(v_i, e_j)$ equals 1 or 0

- Consequently, all vertices in a hyperedge are handled equally; relative "similarity", "affinity", etc. between vertices is discarded

- Leads to loss of some information, which may be harmful to some hypergraph based applications

Probabilistic Hypergraph

- Somehow application dependent
- Depends on the "relationship" itself (and its attributes)
- For instance: assume $|V| \times |V|$ relationship (e.g. similarity, affinity) matrix A over V computed based on some measurement and $A(i,j) \in [0,1]$

  Procedure:
  - Take each vertex as a 'centroid' vertex and form a hyperedge by a centroid and its k-nearest neighbors

    -> the size of a hyperedge is k + 1
  - The incidence matrix H of a probabilistic hypergraph
    - $h(v_i, e_j) = A(j,i)$, if $v_i \in e_j$
    - $h(v_i, e_j) = 0$, otherwise
- In general, assign a probability $P[h(v_i, e_j)]$ s.t. $\Sigma_{i | vi \in ej}\, h(v_i, e_j) = 1$

Probability of Joint failure events

- Individual component failure probability follows a generalized Weibull distribution (with scale parameter b, shape parameter c)
- For component c_i ($1 \le i \le p$)
  - $F_i(t) = \Pr[T_i \le t]$: probability of failure up to time t
  - $R_i(t) = \Pr[T_i > t]$ reliability (or survival) function
- Group comprising p elements survive as none of its individual components fails (assuming dependent failures)
- Generalized multivariate Weibull distribution with joint survival distribution $R_p(t)$

$$\text{Joint survival distr.}: R_p(t) = \exp\left\{ \tau_p^{\nu} - \left[\tau_p + \sum_{i=1}^{p} \lambda_i t_i^{c_i} \right]^{\nu} \right\}$$

where,

λ_i individual failure rates ($\lambda_i > 0$)

τ_p time threshold ($\tau_p \ge 0$)

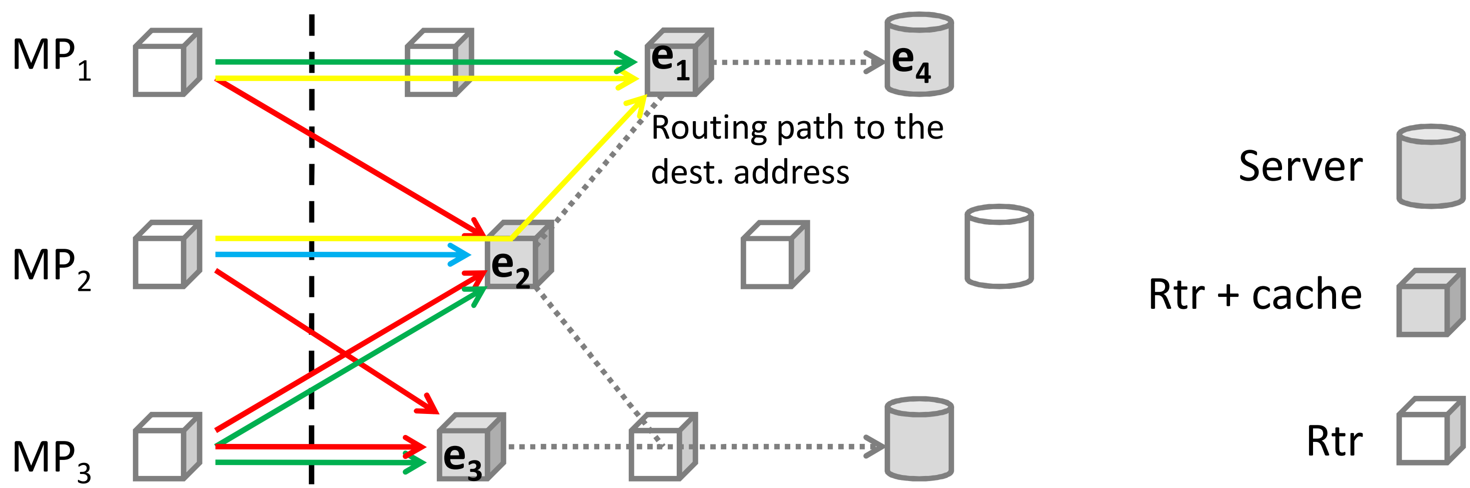
ν coupling effect ($\nu > 0$)

Content networks

- Multiple objects reachable via single address
- Multiple address hosting same object

M:N

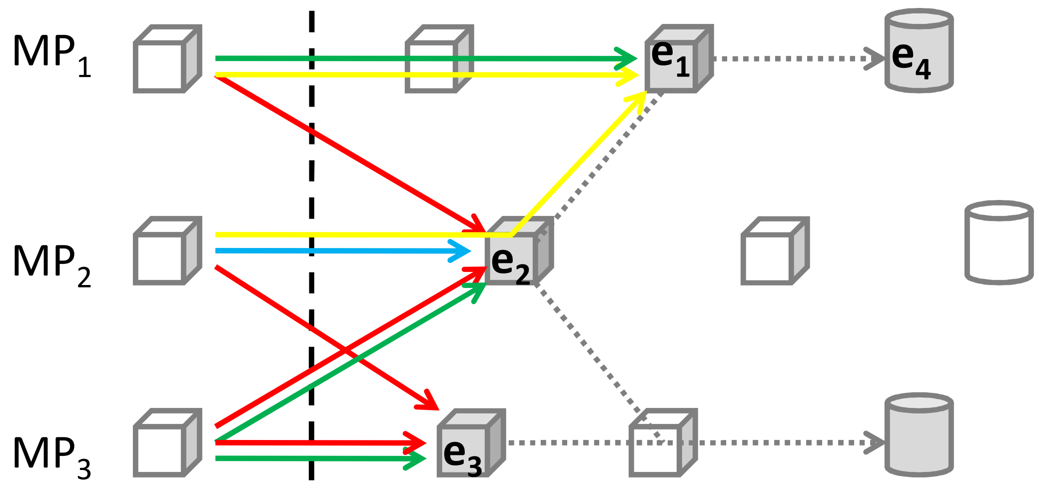
- Example

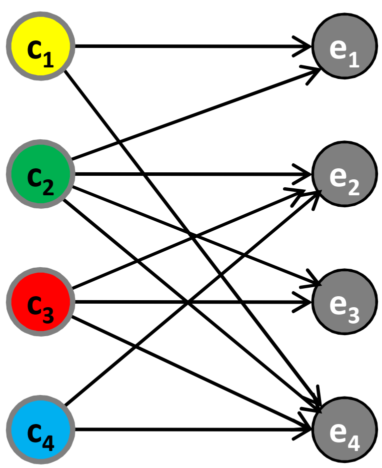


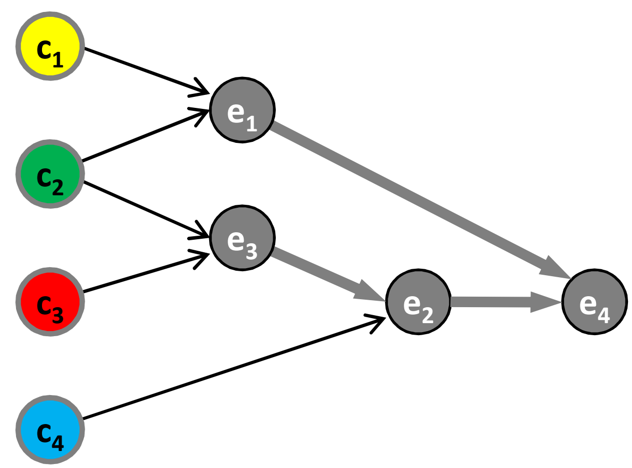
- Objective: MPs to derive the "M:N relationship" (including spatial distribution) from content request/replies

Procedure (example)

- Application of iterative procedure to construct HDAG







Expectations: Hypergraph mining

- Wide space of communication networks applications that can benefit from hypergraph modeling and analysis (not limited to "information systems")

- When involving detection process with uncertainty then probabilistic hypergraphs

- Evolution of networks (programmable networks, in-network caching, etc.) provides additional use cases for "inference"