OBJECT-ORIENTED DESIGN EVALUATION USING ALGEBRAIC GRAPH THEORY

Spiros Xanthos, Alexander Chatzigeorgiou and George Stephanides

Department of Applied Informatics, University of Macedonia, Thessaloniki, Greece

Contact: achat@uom.gr

GOALS

1. To identify and quantify heavily loaded portions of an object-oriented design by algorithmic means
2. To identify dense regions of classes in the system by spectral graph partitioning

"GOD" CLASSES

The outcome of the analysis/design phase is often one or more "God" objects, which perform most of the work in the system. Clearly, such a solution is not managing complexity any better than procedural programming and **implies a poorly designed model.**

Paraphrasing Kleinberg [1]:

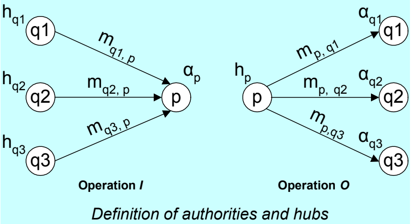
A class c holds a central role in a model:

a. if it receives many messages from other central classes. (candidate of a **good authority**)

b. if it sends many messages to other classes which are also central (candidate of a **good hub**)

LINK ANALYSIS

The set of all classes can be represented as a directed graph $G=(V, E)$ where vertices correspond to the classes and a directed edge indicates an association between classes p and q. Each edge is annotated with the number of discrete messages sent to the same direction ($m_{p,q}$).



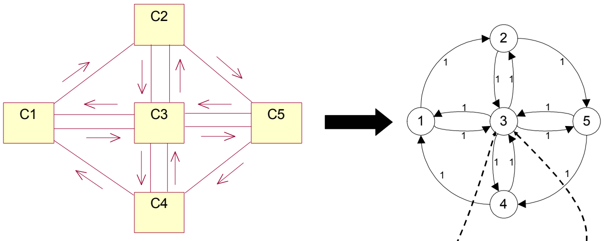
Definition of authorities and hubs

Given a set T of associated classes, the aim is to find the authority and hub weights (a_p, h_p), associated with each class. **Classes with higher authority and hub weights are viewed as classes having a more important role in the model.**

Employing the Power Method of Linear Algebra it can be shown that:

*If A denotes the **adjacency matrix** of the graph G in the model under study, then the authority and hub weights are given by the elements of the normalized **principal eigenvector of*** A^TA ***and*** AA^T*, respectively.*

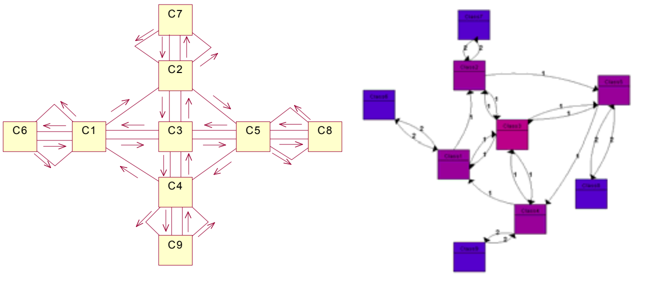
EXAMPLES



$$A = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix} \Rightarrow a_n = \begin{bmatrix} 0.394 \\ 0.394 \\ 0.615 \\ 0.394 \\ 0.394 \end{bmatrix}, \; h_n = \begin{bmatrix} 0.394 \\ 0.394 \\ 0.615 \\ 0.394 \\ 0.394 \end{bmatrix}$$

Principal eigenvectors of A^TA and AA^T

In the following design, other measures (i.e. in-out degree) would not be able to differentiate between the role of the central and peripheral classes.



Colors show the distribution of workload. **Bright red indicates heavily loaded classes**.

Link analysis clearly identifies C3 as the most heavily loaded class:

$$a_n^T = h_n^T = [0.383 \quad 0.383 \quad \boxed{0.454} \quad 0.383 \quad 0.383 \quad 0.227 \quad 0.227 \quad 0.227 \quad 0.227]$$

DENSE COMMUNITIES

The objective is to group classes in such a way that the inter-group coupling is minimized and the intra-group coupling is maximized. Dense communities of classes, might imply relevance of functionality and thus indicate a portion of the overall design that has a distinct purpose.

The second eigenvector of the **Laplacian matrix** of G provides a natural way for bi-partitioning the underlying graph: Positive and negative entries in the second eigenvector correspond to two groups of classes, which, if partitioned, **minimize the coupling between the partitions** [2].

References

[1] J. M. Kleinberg, "Authoritative Sources in a Hyperlinked Environment", *Journal of the ACM*, vol. 46, issue 5, Sep. 1999, pp. 604-632.

[2] M. Fiedler, "A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory," *Czechoslovak Mathematical Journal*, 25 (1975), pp. 619-633.