Spring 2005

The bipartite clique: A topological paradigm for Web user search customization and Web site restructuring

Brenda F. Choyce-Miles

NOTE TO USERS

THE BIPARTITE CLIQUE – A TOPOLOGICAL PARADIGM FOR WEB USER SEARCH CUSTOMIZATION AND WEB SITE RESTRUCTURING

by

Brenda F. Choyce-Miles, BA and MEd

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Computational Analysis and Modeling

COLLEGE OF ENGINEERING AND SCIENCE
LOUISIANA TECH UNIVERSITY

May 2005

INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

LOUISIANA TECH UNIVERSITY

THE GRADUATE SCHOOL

April 27, 2005
Date

We hereby recommend that the dissertation prepared under our supervision by Brenda F. Choyce-Miles entitled The Bipartite Clique – A Topological Paradigm For Web User Search Customization And Website Restructuring be accepted in partial fulfillment of the requirements for the Degree of Doctor (PhD) of Computational Analysis and Modeling

Vir Vrandr Elsha
Supervisor of Dissertation Research

Richard J. Greechie
Head of Department

Engineering and Science
Department

Recommendation concurred in:

Vir Vrandr Elsha

Raja Nassar

Richard J. Greechie

Advisory Committee

Approved:

Director of Graduate Studies

Approved:

Dean of the Graduate School

Dean of the College

GS Form 13
(5/03)

ABSTRACT

The objective of this dissertation research is to aid the Web user to achieve his search objective at a host Web site by organizing a strongly connected neighborhood of Web pages that are thematically and spatially related to the user's search interest. Therefore, methods were developed to (1) find all Web pages at a given Web site that are thematically similar to a user's initial choice of a Web page (selected from the set of Web pages returned in response to a query by any popular search engine), and (2) organize these pages hierarchically in terms of their relevance to the user's initial Web page request. This selection and organization of pages is dynamically adjusted in order to make these methods responsive to the user's choice of pages defining his search agenda.

The methods developed in this work skillfully incorporate the production of the bipartite clique graph structure to simulate both spatial and thematic relatedness of Web pages. By ranking the user's initial page choice as the most relevant page, the *authority* page, link analysis is used to identify a set of pages with out-links to this *authority* page and assemble these into a *hub* of relevant pages. The authority set (initially containing only the user's initial page choice) is then expanded to include other pages with in-links from the set of *hub* pages. The *authority-hub* relationship signified by Web page links is used to define the two partite sets of the biclique graph. The partite set of *authority* pages contains the user's initial page choice and other thematically and spatially similar pages.

The partite set of *hub* pages contains pages whose out-links to the *authority* pages serve as validation of their thematic relevance to the user's search objective.

Two maximal biclique neighborhoods of Web pages specific to the user's interest, containing eight and five pages respectively, were successfully extracted from Web server access logs containing 47,635 entries and 1,140 distinct *request* pages. The iterative use of these methods in association with three Web page metrics introduced in this research facilitated extending a neighborhood dynamically to include nine additional relevant pages.

APPROVAL FOR SCHOLARLY DISSEMINATION

The author grants to the Prescott Memorial Library of Louisiana Tech University the right to reproduce, by appropriate methods, upon request, any or all portions of this Dissertation. It is understood that "proper request" consists of the agreement, on the part of the requesting party, that said reproduction is for his personal use and that subsequent reproduction will not occur without written approval of the author of this Dissertation. Further, any portions of the Dissertation used in books, papers, and other works must be appropriately referenced to this Dissertation.

Finally, the author of this Dissertation reserves the right to publish freely, in the literature, at any time, any or all portions of this Dissertation.

Author Brenda J. Chryce-Miles

Date 4/27/2005

GS Form 14
(5/03)

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ILLUSTRATIONS

GLOSSARY

1. **Adaptive restructuring** – reorganizing or modifying the spatial relationship between web pages at a Web site to reflect users' navigational patterns.

2. **Adaptive Website** - is one that automatically improves its organization and presentation by mining visitors' access data collected in Web server logs.

3. **bipartite clique (biclique) graph** – consists of two partite sets of nodes A and B such that each and every node in A is pair-wise adjacent to each and every node in B, and vice versa. Furthermore, two nodes are adjacent if and only if they are in different partite sets.

4. **bipartite-lattice** – graphically, it is a hierarchal interlacing of two or more biclique subgraphs in such a way that two juxtaposed biclique subgraphs share a common partite set. A biclique-lattice's dimension of level $\lambda = n$ means that n biclique subgraphs are sequentially interlaced.

5. **collective user** – the set of Web users identified by entry data found in the Web server's access log.

6. **Cyber-community** – a large collection of spatially and/or thematically related Web pages/objects that usually span the WWW.

7. **directed (web) graph** ***G(V,E)*** – is composed of a set ***V*** of vertices (URLs ≡ pages or objects), a set ***E*** of edges (links or hyperlinks), and a mapping that assigned each edge an ordered pair of vertices (v_1,v_2) which indicates a directional link from v_1 to v_2.

8. **Host Web site** – the Web site to which a user's requested resource belongs or the Web site to which a user directs his resource accessing activity.

9. **indegree of a node (page)** - the number of arcs whose destination is that node.

10. ***relevant*** **Web page** – "...a WWW page is considered relevant to a query if, by accessing the page, the user can find a resource (URL) containing information pertinent to the query, or the page itself is such a resource"[1].

11. ***referer* (page or node)** – the page containing the outgoing arc whose destination is the requested resource or *request* page (node or page). This page documents how the Web user arrived at a page.

12. ***request* (page or node)** – the page containing the incoming arc from a page called the *referer* page.

13. **URL** – Uniform Resource Locator, the global address of documents and other resources on the World Wide Web.

14. **Web user customization** (preference *or* personalization) – may involve modifying or actually creating Web pages to fit the desires of a particular user, or employing user preference to determine the Web documents to retrieve.

15. **Web log** – a listing of page reference data. It documents user activity at a Web site.

16. **Web mining** – mining data related to the WWW. These data may actually be present in Web pages or it may be related to Web activity, e.g., a Web user's search.

17. **Web neighborhood/community** – a set of web pages associated thematically, contextually, and/or structurally.

18. **Web page** – is a document on the WWW identified by a unique URL. These documents are formatted in a markup language called HyperText Markup Language (HTML) and are accessed via the HTTP communication protocol.

19. **Web structure mining** – performs mining on the links connecting Web pages and objects (video, audio and graphic files).

20. **Web usage mining** – performs mining on data in Web logs. It looks at the history of Web pages visited to identify traffic patterns as these relate to users and the resources accessed.

21. **Web user** – one who accesses and/or utilizes the resources of the WWW.

22. **Web user search** – a goal oriented activity with the objective of locating a specific resource or set of related resources on the WWW.

ACKNOWLEDGMENTS

I am deeply and sincerely grateful to my mentor, Dr. Vir V. Phoha, for his invaluable assistance, direction and encouragement during the course of this work. I take this opportunity to express a special thanks to him for allowing me to be a part of his computer science research group and the confidence he expressed in my ability to make a meaningful contribution in the area of Web mining.

I am indebted to Dr. Richard Greechie for his advisement and guidance during my tenure as an ACAM student.

I dedicate this dissertation to my husband and best friend, Dr. Allen M. Miles, whose patience, support, and confidence in me has never waned.

CHAPTER 1

INTRODUCTION

Web user search customization and the structure of Websites are bound to issues related to optimizing the responsiveness of the World Wide Web (WWW, WWWeb or Web) as an interactive medium. The exponential growth of the WWWeb's sites, pages, and user population is having a decided impact on maintaining the WWW in its role of repository and purveyor of information in this Information Age. The magnitude of information available on the Web and its random unstructured expansion inhibits the locating and ready accessing of information sought by the Web user. For example, the *New York Times* of November 11, 2004 reported that over 8 billion Web pages populate cyberspace [2]. With some three million pages being added daily, it is reported that within ten years most information will be available on the Web, and that 99% of Web information is useless to 99% of Web users [3]. The *abundance problem* and the structural inadequacies characteristic of the majority of Web pages, Websites, and the WWW itself undermine the Web user achieving his *search objective* in an efficient and productive manner without intervention [4, 5].

Search engine technologies, such as Yahoo, Google, Alta Vista and other similar Web services, deliver to the Web user a collection of web pages in which his targeted search object most likely exists. Such collections of web pages are organized as cyber-

communities and are stored in distributed environments [6, 7]. Search results using these technologies can overwhelm a Web user as the choice of a search target object becomes obscured by a large number of related cyber-communities claiming that object. A common experience of most Web users with specific search agendas is the repeated scenario of spending an inordinate amount of time at what initially seems to be a *promising* Website only to find little if anything of relevance, and then moving to the next promising Website only to be forced to continue until he eventually finds the desired page (object) or gives up in frustration. The Web user is thus pushed into the role of *surfer* rather than that of a goal oriented *searcher*.

Web mining research focusing on Web user search customization has been fueled by the recognition that if the WWW is to attain its optimal potential as an interactive medium, the development of new and/or improved information categorization (classification of pages) and retrieval/delivery systems of Web resources organized around user preference is of prime importance [8-10]. The contextual and structural qualities of Web resources have a significant bearing on enhancing the efficiency and productivity of the Web user's search dynamics (navigational pattern).

In response to Web mining issues centered in Web user search customization, this work addresses two specific questions:

(1) How might the Web user be *assisted* in accomplishing his *search objective* in an efficient and productive manner?

(2) How might the Website be made *structurally responsive* to its clientele (users of the site as documented in the Web user access log)?

Thus, the aim of this dissertation research is to develop methods to (1) find all Web pages at a given Website that are thematically similar to a user's initial choice of a Web page (selected from the set of Web pages returned in response to a query by any of the popular search engines), and (2) organize these pages hierarchically in terms of their relevance to the user's initial Web page request. This selection and organization of pages is dynamically adjusted in order to make these methods responsive to the user's choice of pages that define his search agenda.

The methods developed in this work skillfully incorporated the production of the bipartite clique (biclique) graph structure to simulate both spatial and thematic or context relatedness of Web pages. By ranking the initial page choice of the user as the most relevant page, the *authority* page, link analysis is used to identify a set of pages with out-links to this *authority* page and assemble these into a *hub* of relevant pages. The authority set (initially containing the user's initial page choice) is then expanded to include other pages with in-links from the set of hub pages. The *authority-hub* relationship signified by Web page links defines the two partite sets of the biclique graph. The partite set of *authority* pages corresponds to the user's initial choice and other thematically and spatially similar pages, and the partite set of *hub* pages corresponds to those whose out-links to the *authority* pages serves as a validation of their thematic relevance to the Web user's search interest in like manner [11].

The method works as follows. The initial Web page selected by the Web user is made the initial member of the candidate *referer* partite set. All pages with in-coming links of distance one from this initial member page of the candidate *referer* partite set are assigned to the corresponding candidate *request* partite set (hub). In turn, the candidate

referer partite set (authority) is expanded by the assignment of member pages with incoming links of length one from the candidate *request* partite set members. (See Section 3.2.1 for the distance metrics). Thus, the candidate *referer* partite set contains those pages that are most likely to be thematically similar to the user's initial page choice. Four matrices were created to analyze the spatial and thematic qualities between pages belonging to the respective partite sets. The two adjacency matrices (one with candidate *request* page row-headers and candidate *referer* page column-headers, and the other with candidate *referer* page row-headers and candidate *request* page column-headers) were used to identify directional links of length one between two Web pages that are members of the two disjoint candidate partite sets. The cell entry '1' indicates a link from the column entry page to the row entry page, and a '0' means the absence of a link. Matrix operations performed using these two adjacency matrices result in a third adjacency matrix that identifies bi-directional links between members of the two candidate partite sets. The hit frequency matrix's cell entry k at (*page 1, page 2*) means that a count of k distinct accesses of *page 2* were made via *page 1*. By applying various heuristics to the adjacency and hit frequency matrices, two maximal bipartite cliques of relevant pages were extracted. The process is then executed iteratively to generate the biclique-lattice graph that extends the virtual search neighborhood to successive levels. This requires that the current *request* partite set be assigned the role of the *referer* partite set and then repeat the process from the point of instantiating a new candidate *request* partite set based on the user's choice of a page from the new *referer* page set. The process is repeated until no additional pages of relevance can be found or the user terminates his search. These hierarchically organized bicliques (a biclique-lattice) are the final product of the

algorithm ***findBipartiteClique*** (Section 3.3). They represent the virtual search neighborhood of the most relevant Web pages organized hierarchically around the Web user's search agenda.

The organization of this dissertation research is as follows: *A*–Chapter 2 contains a review of literature that highlights the impact that research efforts to improve the interactive nature of the WWW (for the masses or at best a sizable select user population) is having on clearly defining and forcing the issue of *Web user search customization*; *B*–Chapter 3 details the methods developed in this research to enhance Web user search customization. The ***findBipartiteClique*** algorithm introduced in this work is designed to ensure that the user's preference is the driving force behind assembling a virtual bipartite clique search neighborhood that is spatially and thematically consistent with the Web user's search objective. An experiment is conducted using actual Web server (user access) log data to test the theoretical soundness of the approach used in this research; *C*–Chapter 4 highlights the experimental findings of this research relative to the methods' effectiveness in actuating *Web user search customization* via the biclique neighborhood infrastructure, make a pointed recommendation for the formatting of Web server access log data, and suggests a direction for expanding this research.

CHAPTER 2

LITERATURE REVIEW

2.1 User Search Customization

The majority of research centered in Web user customization view the user as a *surfer* with at most a search agenda formed *on-the-fly* or as a *collective entity* with an expansive smorgasbord-like browsing agenda. The user, by way of a list of keywords called descriptors, characterizes the Web resources in which he is interested. The search engine or like services offer to the user *all* pages/objects (URLs) of interest under its purview. A likely scenario is illustrated in Figure 1 where over 400,000 pages/objects were returned in response to the search descriptor '*Web user search customization*' by the Google search engine (**www.google.com**).

Figure 1. Google's response to descriptor '*web user search customization.*'

A Web community is a collection of Web pages that meet a specific set of topic or subject content criteria [3]. The thematic correlation between Web pages that are structurally connected (via hyperlinks) is an established fact [6, 12-14]. Hence, graph

theory lends itself naturally to assembling a community of Web pages that are topically related. However, from the perspective of Web user customization, the quality of any such community is a subjective assessment made by the user who is attributed the status of expert in regards to the selection of pages which defines his Web search agenda [15].

Most research effort is geared toward amassing cyber communities of Web pages of interest within a '*what's related context*' [6, 7, 16]. Page et al. [11] increased the effectiveness and thereby the efficiency of the traditional search engine, which uses key words to identify and cluster Web pages that are topically similar, with the application of their PageRank algorithm which measures the importance of a Web page based on the number of Web pages that point to it. PageRank delivers a cluster of topically similar Web pages returned in response to a Google query that are ranked and ordered by importance [17]. Kumar et al. [6] and Toyoda and Kitsuregawa [18] enhanced an existing related page algorithm to identify Web communities and create a global Web community chart that connects related communities so that "users can navigate through related pages and communities." Their Web community chart of 1,882 related communities was extracted from 17 million Web pages. Broder et al. [19] report experiments on local and global properties of the Web graph using two AltaVista crawls with over 200 million pages and 1.5 billions links each. Analyzing the behavior of Web algorithms that make use of link information, predicting the evolution of web structures such as bipartite core and Web rings, and developing better algorithms for discovering and organizing them are three of the five primary reasons cited for developing and understanding the properties of the Web graph. Kumar et al. [4] and Garofalakis et al. [16] identify Web mining techniques that exploit hyperlink information for discovering

Web structure and Web communities. Kumar et al. [6] use link analysis to mine for global community structures on the Web. They identify instances of bipartite graph structures indicative of emerging communities. Co-citation (URL) is used to establish links between thematically related pages and objects that do not refer to each other directly.

Chen and Park [20] direct their efforts toward extracting large forward sequential traversal patterns from Web access logs in a distributed environment. Notably, many of the pages (objects) defining a *popular* path of a sequential search neighborhood may not be related structurally (link) or content-wise. This approach does not guarantee that intermediate nodes (pages) have anything to do with the user's search objective. The user who embarks upon a search for specific information by traversing a *frequented or popular path* will more than likely arrive at a terminus without achieving his search objective. Web user search customization can be undermined by this approach.

Wang et al. [21] utilized their link analysis algorithm, derived from HITS, on Web access logs to compute Web users' relevance to a given topic. They demonstrated empirically the important connection between the Web user and the Web pages he visits. Their stated findings were that "web-pages and web-users mutually reinforce each other in an iterative way." In regards to advancing *Web user customization*, "... computing the web user's relevance to a given topic is an important task for any *personalization* service on the Web."

The authority-hub relationship between Web pages as expressed by link structure has proven to accurately connote both topical and spatial relatedness [11, 21, 22]. It is

widely used in Web mining research to find and organize global and distributed communities of pages that are thematically similar [6, 11, 18, 23, 24].

Rangarajan et al. [9] and Xie and Phoha [25] cluster Website users into groups based on their Web log access patterns. Their clustering algorithm produces a prototype vector that represents each user cluster by generalizing the URLs most frequently accessed by its members. This prototype vector is used in a prefetching strategy that predicts and delivers pages (objects) sought by the user. The clustering of Web users creates communities of users based on their Web interests and correspondingly links the relevance of pages to respective user groups.

The problem of user search customization tackled from the perspective of *user preference* entails maintaining the user as the director and controller of his search agenda and the expert as to choices of Web resources (pages) he deems relevant to his search agenda [10, 15]. Most research generalizes Web user *personalization* by adopting a development policy of "*what's good for the majority is good for the individual.*" However, Miles and Phoha [23] used Web linkage and usage analysis to identify and rate Web pages that they demonstrate graphically to be strongly connected to the *individual* Web user's interest (as indicated by his choice of a Web page) and assemble these into a topologically cohesive community of pages hierarchically arranged around the interest of the Web user. They use the collective user's access history of these pages of interest in a collaborative manner that allows their approach to take advantage of the expertise of all other users whose search agendas were similar to the current Web user. Srivastava et al. [26] propose a taxonomy of Web usage mining applications that places personalization or Web user customization at the top.

2.2 User Directed Website Restructuring

Graph theory presents itself as a natural analytical tool to study both structure and function of Web user search dynamics. Graphically, the WWW itself has been documented to adhere to a sparse random graph generation dynamic in which nodes (Web pages/objects) are added, deleted, and edge assignments (directed links between pages/objects) between nodes may be new, copied, or nil [4, 19]. Although the overall topology of the WWW with its billions of nodes exhibits little or no structure, some very important thematically cohesive regions of the Web have presented themselves. The several hundred thousand instances of disjoint bipartite cores found to exist on the Web pinpoint the bipartite as a significant Web subgraph. The topological characteristics of the bipartite core has been used to identify existing and emerging cyber-communities of Web pages/objects [6, 19].

Dill et al. [27] report on the *self-similarity* characteristic in the WWW. The import of *self-similarity* is that "each thematically unified region displays the same characteristics as the Web at large." Although the *unified* region in their work is made up of a cluster of Web pages sharing a common syntactic trait (URL indexing) and distributed across the WWW, the approach in this work, for the purpose of Web user search customization, restricts the selection of Web pages to a single Website and the creation of a unified region of pages relevant to one Web user's objective. The theoretical concept of self-similarity formulated by Dill et al. can logically be applied to a single Website as indicated by their observation that "...at various different scales, cohesive collections of Web pages (for instances, pages on a site or pages about a topic) mirror the structure of the Web at-large." In this dissertation research, the bipartite clique

(biclique) graph structure of a virtual search community of Web pages is the infrastructure of choice for establishing criteria to create a thematically unified region specific to a Web user's preference [23].

Restructuring activities essential to helping the Web user identify and gain access to pages of relevance (pages germane to his search objective) populating this spatially and thematically cohesive region are needed. Web user search customization requires that a Website be responsive to the legitimate demands of its clientele in regard to page content, location, and presentation [21, 25, 28]. An example of a legitimate demand is the request made by the Web user for a particular page whose existence is acknowledged in the text of a page, but no link or access path is indicated or apparent to the Web user. Adaptive restructuring can be used to improve the organization and presentation of a Website by mining visitor access data collected in Web server logs [28]. Since the approach used in this work creates a *virtual* search neighborhood at the Web user's behest guided by criteria of search legitimacy, linkage analysis is performed to install the *request* page in its optimum position in the virtual biclique search neighborhood based on the level of its relevance to the user's search agenda as purported by the collective user's history of traversal patterns around the page. Unlike adaptive restructuring, this approach leaves the actual Website intact.

The knowledge base formulated upon the thesis of maintaining the spatial and thematic organization of pages populating the virtual biclique search neighborhood can be mined for traversal patterns that yield Website restructuring information that can improve the site's responsiveness to its users [23]. Xiao and Dunham [29] mined Web

server logs for traversal patterns of Web users to uncover information about the site's services that could be used to improve its design.

CHAPTER 3

METHODOLOGY

The approach used in this work to achieve *Web user search customization* is novel in its use of the biclique infrastructure to hierarchically organize a strongly connected neighborhood at a host Website and populate it with thematically related pages specific to a Web user's search agenda. Furthermore, it is innovative in its introduction of new metrics to gauge the quality of pages and the structural impact of the placement of pages used to populate the biclique search neighborhood [30].

In the rest of the chapter, the methodology is presented in the following order: (1) description of the data set, (2) introduction to three new Web metrics, (3) the design of algorithm ***findBipartiteClique (fBC)***, and (4) Experiment-Test algorithmic methods: *i*) Phase I. Generate level-1 biclique, *ii*) Phase II. Generate level-2 biclique [Iterative use of algorithm ***fBC*** to produce biclique-lattice], *iii*) Phase III. Implement ***tri-paginal*** metric for restructuring activities of add/delete links, rate fitness of a candidate request page, and qualify pages of value for membership in the *request* partite set, *iv*) Phase IV. Generate level-2 biclique-lattice using restructuring.

3.1 The Data Set

The Web access logs used in this work were selected from the Music Machines' Website archives which were collected from 2/12/1997 through 4/30/1999. Each log in the archive contains user access data collected over a distinct 24-hour (day) period. These data were stored in the Combined Log Format (C_bLF). These logs are publicly available at **http://www.web-caching.com/traces-logs.html**.

The experiment conducted in this research used five of the February 1997 archive logs. The experimental data set ***D*** contains the combined total of 46,735 user access entries from these five logs. Visual FoxPro 7.0 Database System was used to collect, clean, organize, and query the log data for this experiment.

Most Web servers store their user access data in the Common Log Format (CLF) which uses parameters 1 to 7 in Table 1. The C_bLF is the same as the CLF with the addition of the *referer* and *user_agent* parameters [31]. The C_bLF access entries conform to the string expression "***$host*** + ***$ident_result*** + ***$auth_user*** + ***$date_time*** + ***$request*** + ***$status_code*** + ***$bytes_sent*** + ***$referer*** + ***$user_gent.***" The C_bLF's use of the *referer* parameter yields page linkage data that is a boon to Web structure mining in that the log entry not only identifies the *request* page but also the page from which the *request* page was directly accessed.

Table 1. Web server 's user access log parameters for the common and combined log formats.

#	Parameter	Description	Example
1	*host*	IP address or hostname of the remote user requesting the page.	203.99.222.999 or latech.edu
2	*Ident_result*	Field used to log response returned by remote user's identd server.	not used usually; "-" inserted
3	*authuser*	'ecHTTP' authentication used to restrict access to some of user's web documents.	username of authenticated user for transaction
4	*Date_ time*	Date and time (24-hour format and time zone) of user's access.	[12/Jun/1997:00:09: 12-0700]
5	*request*	Actual request sent by remote user.	"GET/HTTP/1.0"
6	*Status_code*	Three digit number returned by the server that specifies how the request was handled.	200- handled okay 404- document not found 500- server-side error
7	*Bytes (sent)*	Amount of data returned by server.	1523
8	*referer*	The referring page as reported by the remote user's browser. (This is the standard spelling in the HTTP specification and the environment variable).	HTTP specification
9	*User_ agent*	The type and version of browser software used as reported by the remote user's browser.	Microsoft Internet Explorer 6.02

The Music Machines logs were preprocessed so that its user access entries conform to the string expression "***$host*** + ***$date_time*** + ***$request*** + ***$referer***" which are parameters 1, 4, 5, and 8 of Table 1. Each host name was replaced with a unique integer id to maintain user privacy rights. Entries in the prepossessed log contain four fields: (1) ***Origin***, the host id of the originator of the request, (2) ***Time***, the timestamp of the request, (3) ***URL***, the page requested and (4) ***Referer***, the referring page. Illustration 1 shows four actual log entries from a segment of one user session in the archives. Fields are delimited by '||.'

Illustration 1. Four log entries in Music Machine archive log of 2/16/1997.

Origin **(O) || *Time* (T) || *Request* (U)|| (R)**

O:00004233 || **T**:1997/02/16-11:47:02 || **U**:/music/machines/||
R:http://www.hyperreal.com/

O:00004233||**T**:1997/02/16-11:47:12 || **U**:/music/machines/guide/||
R:http://www.hyperreal.com/music/machines/

O:00004233||**T**:1997/02/16-11:47:24||**U**:/music/machines/manufacturers/||
R:http://www.hyperreal.com/music/machines/guide/

O:00004233||**T**:1997/02/16-1:47:30||**U**:/music/machines/manufacturers/Emu/||
R:http://www.hyperreal.com/music/machines/manufacturers/

O:00004233||**T**:1997/02/16-11:47:43||
U:/music/machines/manufacturers/Emu/Emulator/||
R:http://www.hyperreal.com/music/machines/manufacturers/Emu/

Illustration 2. Music Machine user session O:0665 excerpted from Web server access log dated 2/12/1977.

```
Origin:O:0665 Date:2/12/1997
URL-Request
  ||
=============================================================

U:/music/machines/
|| R:http://www.harmony-central.com/Links/#lists

U:/music/machines/search.html
|| R:http://www.hyperreal.com/music/machines/

U:/music/machines/manufacturers/
|| R:http://www.hyperreal.com/music/machines/search.html

U:/music/machines/manufacturers/Studio-Electronics/
|| R:http://www.hyperreal.com/music/machines/manufacturers/

U:/music/machines/manufacturers/Studio-Electronics/images/
|| R:http://www.hyperreal.com/music/machines/manufacturers/Studio-
Electronics/

U:/music/machines/manufacturers/Studio-Electronics/images/
|| R:http://www.hyperreal.com/music/machines/manufacturers/Studio-
Electronics/

U:/music/machines/manufacturers/Studio-Electronics/
|| R:http://www.hyperreal.com/music/machines/manufacturers/

U:/music/machines/manufacturers/Studio-Electronics/info/
|| R:http://www.hyperreal.com/music/machines/manufacturers/Studio-
Electronics/

U:/music/machines/manufacturers/Studio-Electronics/
|| R:http://www.hyperreal.com/music/machines/manufacturers/

U:/music/machines/manufacturers/
|| R:http://www.hyperreal.com/music/machines/manufacturers/Studio-
Electronics/

U:/music/machines/manufacturers/Mackie/
|| R:http://www.hyperreal.com/music/machines/manufacturers/

U:/music/machines/manufacturers/Mackie/MS-1202/
|| R:http://www.hyperreal.com/music/machines/manufacturers/Mackie/

U:/music/machines/manufacturers/Mackie/MS-1202/
|| R:http://www.hyperreal.com/music/machines/manufacturers/Mackie/

U:/music/machines/manufacturers/Mackie/   ||
R:http://www.hyperreal.com/music/machines/manufacturers/
```

Figure 2 graphically depicts the page link defined by a log entry in C_BLF. The first two entries in Illustration 2 are defined by replacing the ***Referer*** and ***Request*** labeled pages in Figure 2 with the 'U' and 'R' entries in Illustration 1 respectively. The directed link between *referer* (authority) and request (hub) is inherently definitive of adjacency of a spatial and contextual (topical) nature. The length of edge (***Referer, Request***) connecting the two pages is ***d***. The distance between two pages from different Websites is defined in this work as infinity.

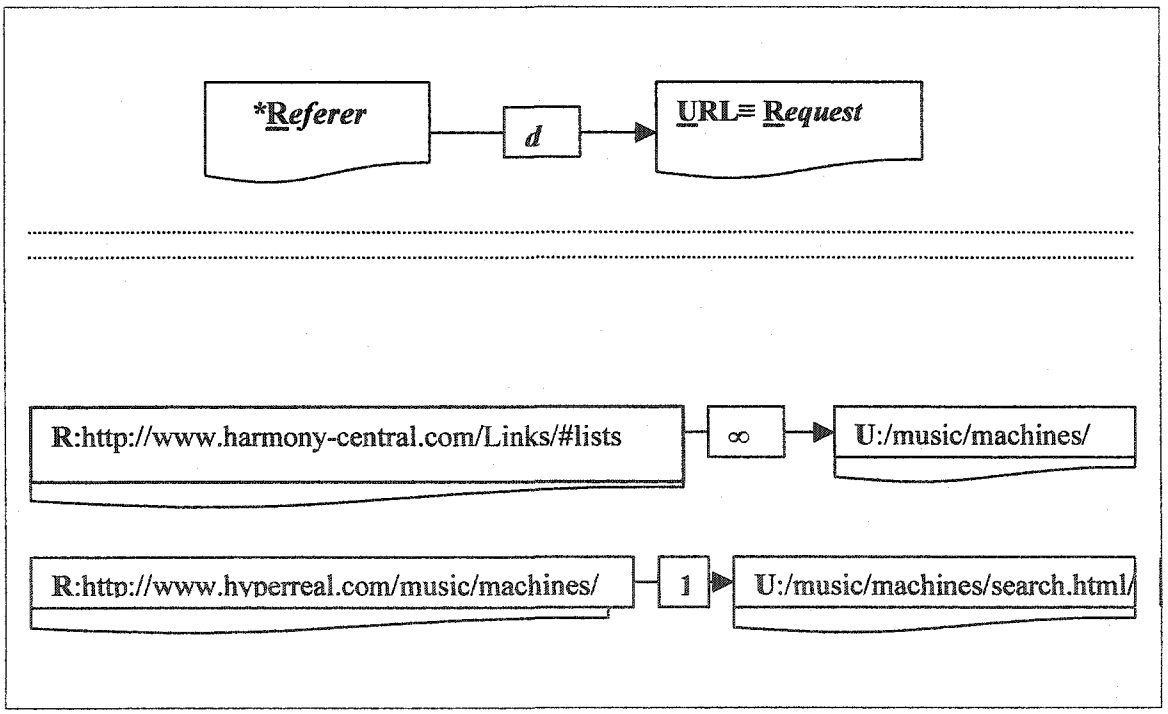

Figure 2. The directed graph representing *referer* to *request* link defined in a C_bLF access entry.

3.2 Web Page Relevance Metrics

In this dissertation research, three new metrics were designed to measure (1) the *distance* between two Web pages, (2) the *interest level* of a page relative to other pages already identified as high quality Web pages of value, and (3) the *fitness* of a Web page to occupy a position of relevance in a search neighborhood.

Germane to understanding these metrics is a very basic understanding of what a Web page is. A Web page is a document on the WWW that is identified by a unique Uniform Resource Locator (URL, the global address of documents and other resources on the WWW). URL http://www.hyperreal.com/music/machines/manufacturers is a page taken from data set ***D*** used in this work. For indexing, referencing, and measuring Web page attributes, 'http://www.n_1/n_2/…/n_k' is the general format for the URL of a Web page. Hence, the URL from data set ***D*** has four nodes (n_1='hy*perreal.com*', n_2='mu*sic*', n_3='*machines*', n_4='manufacturers').

Let $\mathbf{P_n}$, $\mathbf{P_j}$ and $\mathbf{P_k}$ be pages hosted by the same Website. They have all nodes **n** in common up to and including node $\mathbf{r_i}$ with $\mathbf{P_n} \neq \mathbf{P_j} \neq \mathbf{P_k}$. Their respective URL's are:

$\mathbf{P_n}$= http://www.n_1/n_2/…/n_{i-1}/$\mathbf{r_i}$/,

$\mathbf{P_j}$ = http://www.n_1/n_2/…/n_{i-1}/$\mathbf{r_i}$/$\mathbf{a_{i+1}}$/…/$\mathbf{a_{i+(N1-1)}}$/$\mathbf{a_{i+N1}}$/ and

$\mathbf{P_k}$= http://www.n_1/n_2/…/n_{i-1}/$\mathbf{r_i}$/$\mathbf{b_{i+1}}$/…/$\mathbf{b_{i+(N1-1)}}$/$\mathbf{b_{i+N2}}$/

where $\mathbf{i} \geq 2$ for i, N_1, $N_2 \in \{1, 2, 3, \ldots\}$. The node sequence $S_N(P)$ of each page is represented by:

$S_N(\mathbf{P_n})$ = "n_1/n_2/…/n_{i-1}/$\mathbf{r_i}$/,"

$S_N(\mathbf{P_j})$ = "n_1/n_2/…/n_{i-1}/r_i/a_{i+1}/…/$a_{i+(N1-1)}$/a_{i+N1}/" and

$S_N(\mathbf{P_k})$ = "n_1/n_2/…/n_{i-1}/r_i/b_{i+1}/…/$b_{i+(N2-1)}$/b_{i+N2}/."

3.2.1 Distance Metrics

α-distance. If a page P's node sequence $S_N(P)$ can be expressed as the node sequence of another page concatenated (+) with a node sub-sequence of P, then α-distance is defined between these two pages. Consider the node sequence of page $\mathbf{P_j}$.

$$S_N(\mathbf{P_j}) = \text{`}n_1/n_2/\ldots/n_{i-1}/r_i/a_{i+1}/\ldots/a_{i+(N1-1)}/a_{i+N1}/\text{'}$$
$$= \text{`}n_1/n_2/\ldots/n_{i-1}/r_i/\text{'} + \text{`}a_{i+1}/\ldots/a_{i+(N1-1)}/a_{i+N1}/\text{'}$$
$$= S_N(\mathbf{P_n}) + \text{`}a_{i+1}/\ldots/a_{i+(N1-1)}/a_{i+N1}/\text{'}$$

Therefore, $d_\alpha(P_n,P_j)=(i+N1)-i = N1$. Note also that $d_\alpha(P_n,P_k)= (i+N2)-i = N2$, whereas $d_\alpha(P_j, P_k)$ is not defined.

β-distance. If α-distance is *not* defined for two pages, then

$$d_\beta(P_j, P_k) = d_\alpha(P_n, P_j) + d_\alpha(P_n, P_k) - 1 = N1 + N2 - 1.$$

∞-distance. If pages $\mathbf{P_x}$ and $\mathbf{P_y}$ belong to different Websites, then

$$d(\mathbf{P_x},\mathbf{P_y}) = d(\mathbf{P_x},\mathbf{P_y}) = \infty.$$

α -adjacent. Two pages $\mathbf{P_i}$ and $\mathbf{P_j}$ are αlpha-adjacent (α-adj) if and only if edges $(\mathbf{P_i},\mathbf{P_j})$ and $(\mathbf{P_j},\mathbf{P_i})$ exist (i.e., $\mathbf{P_i} \rightarrow \mathbf{P_j}$ and $\mathbf{P_j} \rightarrow \mathbf{P_i}$ means $\mathbf{P_i} \leftrightarrow \mathbf{P_j}$) and

$$d(\mathbf{P_i},\mathbf{P_j}) = d(\mathbf{P_j},\mathbf{P_i}) = 1.$$

3.2.2 Minimal Interest Level Metric

This measurement allows a page p_i that has been implicated as having relevance, although having a lower hit frequency than its thematically similar counter-parts, to be re-assessed in comparison to the mean hit frequencies of its counter-parts. This metric is designed to minimize the elimination of thematically valuable pages from the search neighborhood by widening the statistical rating range for relevance. The **Minimal Interest Level (MinIL)** metric is presented in Illustration 5 of Section 3.3.

3.2.3 Tri-paginal Link Metric

The tri-pagninal link (**3-*pL***) structure ($\mathbf{p_i,p_n,p_j}$) pictured in Figure 3 is the unit structure of the biclique infrastructure of the search neighborhood. The tri-paginal link meter (**3-*pLM***) is a design tool used to ensure the structural integrity of the virtual biclique neighborhood generated around user preference. This tool is essential to the restructuring activity of adding links to extend the biclique search neighborhood. First, the **3-*pLM*** measures the *fitness* of a page for inclusion in the extended neighborhood. Secondly, if the fitness rating indicates current relevance, the metric is used to identify specific criteria for fully qualifying the page for inclusion in the *hub* partite set of the biclique subgraph.

The tri-paginal link, ***3-pL*** $\equiv (\mathbf{p_1, p_2, p_3})$, is introduced in this work as the structural unit of the biclique-lattice generated via the iterative use of the ***fBC*** algorithm. This structural metric gauges the spatial qualities of two authority pages relative to a third page called the nucleus (a hub page) for **α**-adjacencies (i.e., the bi-directional links defined by $\mathbf{p_1 \leftrightarrow p_2 \leftrightarrow p_3}$). The tri-paginal link metric's implementation as a spatial gauge for the biclique infrastructure is designed to identify and/or measure:

- links for page adjacency between candidate pages of the respective partite sets,
- hub-authority dynamics between pages qualified for inclusion in a level $\lambda > 1$ biclique subgraph, and
- the fitness (relevance) of each potential *hub* page (user preference moderates this assignment and rating) and qualify or disqualify these pages for inclusion in the *hub* partite set. By this, metric criteria are established for add-link restructuring activities that make pages of relevance accessible to the Web user dynamically.

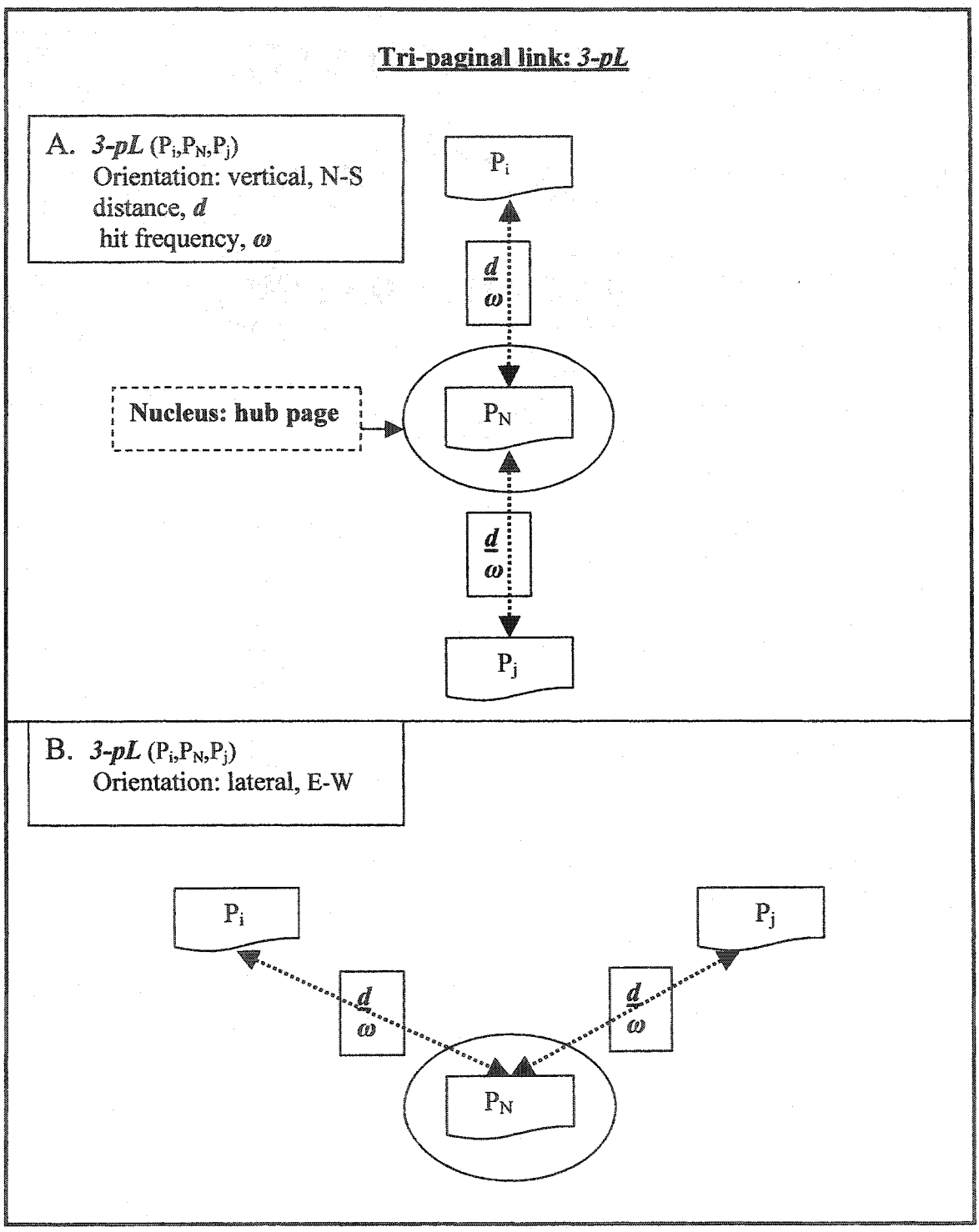

Figure 3. The 3-pL is the unit structure of the biclique infrastructure of the search neighborhood.

3.3 The Algorithm–*findBipartiteClique*

Algorithm ***findBipartiteClique***[Website ***Ws***, page r_i], denoted (***fBC***), is designed to generate a virtual biclique search neighborhood populated with pages relevant to the user's initial Web page choice r_i which is used to characterize the user's search interest and also define his search agenda. This procedure builds a biclique search neighborhood organized around user preference by employing authority-hub dynamics to identify and partition relevant Web pages hosted by Website ***Ws*** into either the *referer* or the *request* partite set.

3.3.1 Pseudo-code for *fBC*

The ***fBC*** algorithm is presented in Illustration 3, and a demonstration of its use follows in Illustration 4.

Illustration 3. Algorithm: *findBipartiteClique* [Website Ws, page r_i] generates a virtual biclique neighborhood of Web pages rooted in user preference r_i.

Objective : To assemble a bipartite clique (graph) of Web pages specific to user preference by employing the cumulative history of users with similar search agendas collaboratively. Note: The C_bLF Web server access log structure is required.

Input : r_I, initial page or object representing Web user preference or search interest.

Output : *bipartite clique*, strongly connected neighborhood of relevant pages.

---

1 ***Input*** **initial** r_1 //* *Web user selected page, URL ε Website Ws with C_bLF access log structure* */*

2 ***Assign*** r_1 **to partite set** R_F //* *biclique characterized as rooted in* r_1 */*

Algorithm *fBC* continued on next page

3 *Determine* $CR_Q = \{p_{rj} \mid r_I \rightarrow p_{rj}$ **and** $d(r_I, p_{rj}) = 1\}$ *//* Candidate request partite set assigned Web pages *//*

4 *Set* $m = \#(CR_Q)$ *//* cardinality of* CR_Q **//*

/ Expand candidate partite set* CR_F. **/*

5 *For* **each page** $p_{rj} \varepsilon CR_Q$ $(1 \leq j \leq m)$ */* repeat 5.1 m times */*

/ Calculate set* R_j *of pages with in-links from* p_{rj}, *referred to by request* p_{rj}. **/*

5.1 $R_j \equiv \{ r_{jk} \mid p_{rj} \rightarrow r_{jk}$ **and** $d(p_{rj}, r_{jk}) = 1\}$

6 $CR_F = U_{(1 \leq j \leq m)} R_j$ */* candidate partite set defined */*

7 *Set* $N = \#(CR_F)$ */* cardinality of* CR_F **/*

/ Instantiate m-by-*N *adjacency matrix* $A_{rf \rightarrow rq} \equiv CR_Q \times CR_F$ *where edge* $rf_f \rightarrow rq_q$ *exist and* $d(rf_f, rq_q) = 1$, *otherwise* $d(rf_f, rq_q) = 0$. **/*

8 **DO PROCEDURE** *Instantiate*_A $[A_{rf \rightarrow rq}]$

/ Apply Preprocessing criteria to adjacency matrix* $A_{rf \rightarrow rq}$ **/*

9 *Set* $CR_F = CR_F - \{rf \mid$ *referer rf* **references < 45% of requests** $rq \varepsilon CR_Q\}$

10 $N = \#(CR_F)$ */* Reset N ; size of CRF is reduced by preprocessing activity */*

/ m-by-*N *hit frequency matrix* $H_{rf \rightarrow rq} = [(\omega(rf_c, rq_r))_{rc}] \equiv CR_Q \times CR_F$ *assigned edge count of edges* $(rf_c, rq_r) \equiv rf_c \rightarrow rq_r$ **for** $rf_c \varepsilon CR_F$ $rq_r \varepsilon CR_Q$ **/*

11 **DO PROCEDURE** *Instantiate*_H $[H_{rf \rightarrow rq}]$

/* Apply *metric: Minimum Interest Level* **preprocessing criteria to matrix** $H_{rf \rightarrow rq} = [(\omega(rf_c, rq_r))_{rc}]$; *Method minIL execution is invoke in 12 */*

12 $CR_Q = CR_Q - \{ rq_r \mid$ **minIL**$[K_1, K_2, medhf, rq] = 0\}$

13 $N = \#(CR_F)$

Algorithm *fBC* continued on next page

/* *Convert hit frequency matrix* $H_{rf \rightarrow rq} = [(\omega(rf_c, rq_r))_{rc}] \equiv CR_Q \times CR_F$ *to an adjacency matrix* ω *replaced by adjacency indicator 1* $\equiv$ *edge* $rf_f \rightarrow p_q$ *exist and* $d(rf_f, p_q) = 1$, otherwise $d(rf_f, p_q) = 0$. */

14 *For* f = 1 *to* m /* *column index s* */

15 *For* q = 1 *to* N /* *q is row index of request pages* */

16 *if* $(\omega(rf_f, p_q))_{qf} \geq 1$

17 $(\omega(rf_f, p_q))_{qf} = 1$ /* *link* $rf_f \rightarrow p_q$ *exists and adjacency indicator is set to 1* */

18 *else*

19 $(\omega(rf_f, p_q))_{qf} = 0$ /* *link* $rf_f \rightarrow p_q$ *does not exist and adjacency indicator is set to 0* */

20 End_*if*

21 *End_For*

22 *End_For*

/* ***Interchanging row and column header assignments and edge page components to indicate the directional change of page link the N-by-m adjacency matrix*** $A_{rf \leftarrow rq} \equiv CR_F \leftarrow CR_Q$ ***is instantiated .***/

23 **DO PROCEDURE** *Instantiate_A* $[A_{rf \leftarrow rq}]$

/* *replacing* ω *values with adjacency indicator, 1* $\equiv$ *edge* $rq_q \rightarrow rf_f$ *exist and* $d(rq_q, rf_f) = 1$, otherwise 0).

/* *Use matrix addition on adjacency matrices* $A_{Q \leftarrow F}$ *and* $A_{F \leftarrow Q}$ *to instantiate bi-directional adjacency matrix* $A_{Q \leftrightarrow F}$; $CR_Q \leftrightarrow CR_F$ */

24 $A_{Q \leftrightarrow F} = A_{Q \leftarrow F} - (A_{F \leftarrow Q})^T$ /* *matrix operations* */

/* *replace values with bi-directional adjacency-link indicator:*

0 $\equiv p_q \leftrightarrow r_f$, *i.e., edges* (p_q, r_f) *and* (r_f, p_q) *exist,*
*0** $\equiv$ *neither edge exists*
1 $\equiv p_q \rightarrow r_f$, *i.e., edge* (p_q, r_f) *exists but edge* (r_f, p_q) *does not* */
-1 $\equiv rf_f \rightarrow p_q$, *i.e., edge* (r_f, p_q) *exists but edge* (p_q, r_f) *does not.* */

Algorithm *fBC* continued on next page

/* *Apply preprocessing criteria in Step 9 to bi-directional adjacency matrix* $A_{Q\leftrightarrow F}$ */

25 $CR_F = CR_F - \{ rf \mid$ *referer* rf **references < 45% of requests** $rq \in CR_Q \}$

26 **Set** $R_F = CR_F$

/* *Extract biclique* B_F *that maximizes the referer partite set* R_F */
//* *Maximize the referer partite set : Eliminate from* CR_Q *any request with at least one zero row entry,* ZR_F*. *//

27 *Set* $R_Q = CR_Q - ZR_Q$*

28 **Output biclique** $B_F(R_F \cup R_Q, rf \leftrightarrow rq$

/* *Extract biclique* B_Q *that maximizes the request partite set* R_Q */

29 **Set** $R_Q = CR_Q$*

30 **Set** $R_F = CR_F - ZR_F$* //* *Eliminate from* CR_F *any referers with at least one zero column entry.* *//

31 **Output biclique** $B_Q(R_F \cup R_Q, rf \leftrightarrow rq)$

32 **END-Algorithm** ***fBC.***

***PROCEDURES* called by *fBC* to generate hit frequency and adjacency matrices:**

/* **Instantiate** *m-by-*N *adjacency matrix* $A_{rf\rightarrow rq} \equiv CR_Q \times CR_F$
where edge $rf_f \rightarrow rq_q$ *exist and* $d(rf_f, rq_q)=1$, **otherwise** $d(rf_f, rq_q)=0$. */

```
33  PROCEDURE Instantiate_A [A_rf→rq ≡ CR_Q X CR_F]
34     For c = 1 to m                  /*  column index of referers   */
35        For r = 1 to N               /*  row index of requests      */
36           if   d(rf_c, rq_r) =1
37              A[r][c] = 1     /*  link rf_c→ rq_r exist and d(rf_c, rq_r)=1   */
38           else
39              A[r][c] = 0     /*  link rf_c→ rq_r does not exist or d(rf_c, rq_r)=1  */
40           End_if
41        End_For
42     End_For
43  END PROCEDURE Instantiate_A ----------
```

---------- Algorithm *fBC* continued on next page----------

```
/*  m-by-N hit frequency matrix H_{rf→rq} =[(ω(rf_c, rq_r))_rc] ≡ CRQ X CRF assigned
    edge count of edges (rf_c, rq_r)≡ rf_c → rq_r for rf_c ε CRF rq_r ε CRQ        */

 44  PROCEDURE Instantiate_H [H_{rf→rq} ≡ CRQ X CRF]
 45    For c = 1 to m              /*  column index of referers                   */
 46      For r = 1 to N            /*  row index of requests                      */
 47        H ≡ [(ω(rf_c, rq_r))_rc]  /* ω(rf_c, rq_r) = k means rf_c refers to
                                       request rq_r k times                       */
 48      End-For
 49    End_For
 50  End PROCEDURE Instantiate_H ----------------
/*************************************************************************/

/* Method: Minimum Interest Level (MinIL) metric detects a request page's relevance
in comparison with other request pages that have been verified as pages of relevance
to the user's interest. */

 51  Method minIL[K1, K2, medhf, rq]  /* Minimal Interest Level Detector */

 /* ω ≡ hit frequency
    medhf= Median{ω(rf, rq)} ≡ Median hit frequency of a request page rq.
    rf ≡ referer page
    rq ≡ request page
    K1 ≡ #(referers in CRF that refer to page rq),
    K2 ≡ #(referers in CRF).                                          */

 52   If  ( K1/K2 * Median{ω(rf, rq)} ≥ Median{ω(rf, rq) } )
 53       minIL[rq] = 1;  /* interest detected */
 54     else
 55       minIL[rq] = 0;  /*no interest detected */
 56       eliminate[rq]
 57   End_If
 58  END METHOD-MinIL

END of Algorithm fBC.
```

Assume that the Web user starts his search with URL $r_1 \varepsilon W_s$. The search is thus characterized as rooted in the user's initial page choice r_1. A detailed descriptive walk-through of algorithm *fBC* follows in Illustration 4.

Illustration 4. A walk-through of algorithm *findBipartiteClique*.

Step 1 The Web user initiates his search at Website W_s by his choice of URL r_l.

Step 2 Page r_l is assigned to the *referer* partite set.

Step 3 All pages p_{rj} with incoming links from r_l and $d(r_l, p_{rj}) = 1$ are assigned to the candidate *request* partite set CR_Q. The set of edges $E = \{(r_l, p_{rj}) \mid 1 \le j \le m\}$ in Figure 4 illustrates the m distinct outgoing links from *referer* r_l to each page in CR_Q. Edge (r_l, p_{ij}), symbolized by $r_l \rightarrow p_{rl}$, is defined by C_bLF access entry

'MYHOST.NET||02//12/1999:05:34:25]|| "GET/<p_{rl}>"||"http:/r_l."'

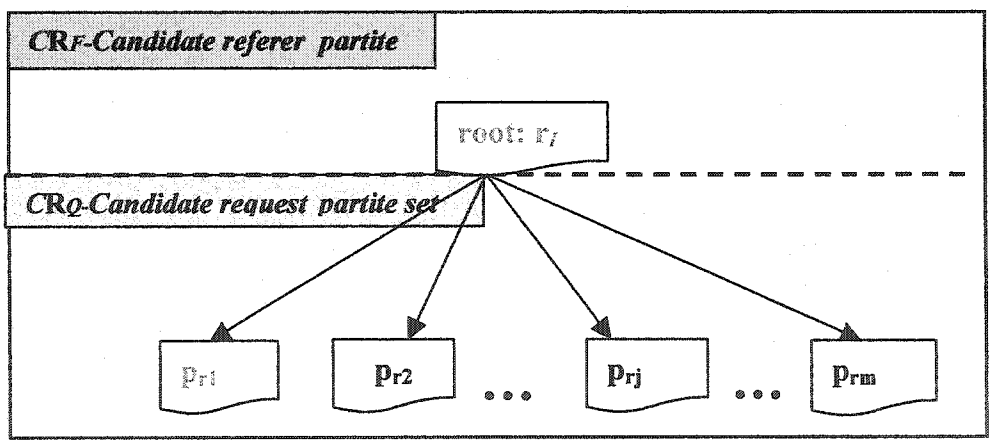

Figure 4. Candidate biclique shows page r_l with m outgoing links.

Step 4 Assign to m the number of pages referred to by r_l.

Step 5 The CR_Q is used to moderate the expansion of the CR_F. For each *request* page $p_{rj} \in CR_Q$, R_i is assigned all pages r_{ij} where link $p_{rj} \rightarrow r_{ij}$ exists and $d(p_{rj}, r_{ij}) = 1$. The set R_i contains all page requests accessed via p_{rj} in W_S log. The mapping

$L(p_i) = R_i = \{ r_{i1}, r_{i2}, r_{i3}, \ldots, r_{ik}^{i} \}$ defines the m respective sets R_i used to expand the candidate *referer* partite set. Hence,

$$L(p_{r1}) = R_1 = \{r_{11}, r_{12}, r_{13}, \ldots, r_{1k}^{1}\},$$

$$L(p_{r2}) = R_2 = \{r_{21}, r_{22}, r_{23}, \ldots, r_{2k}^{2}\},$$

$$\ldots$$

$$L(p_{rm}) = R_m = \{ r_{m1}, r_{m2}, r_{m3}, \ldots, r_{mk}^{m} \}.$$

The set R_j contains all pages r_{jk} with *referer* p_{rj}. That is, page $r_{jk} \in R_j$ means graphically that edge (p_j, r_{ik}) exists and its length d equals 1. The results of Step 5 are illustrated in Figure 5.

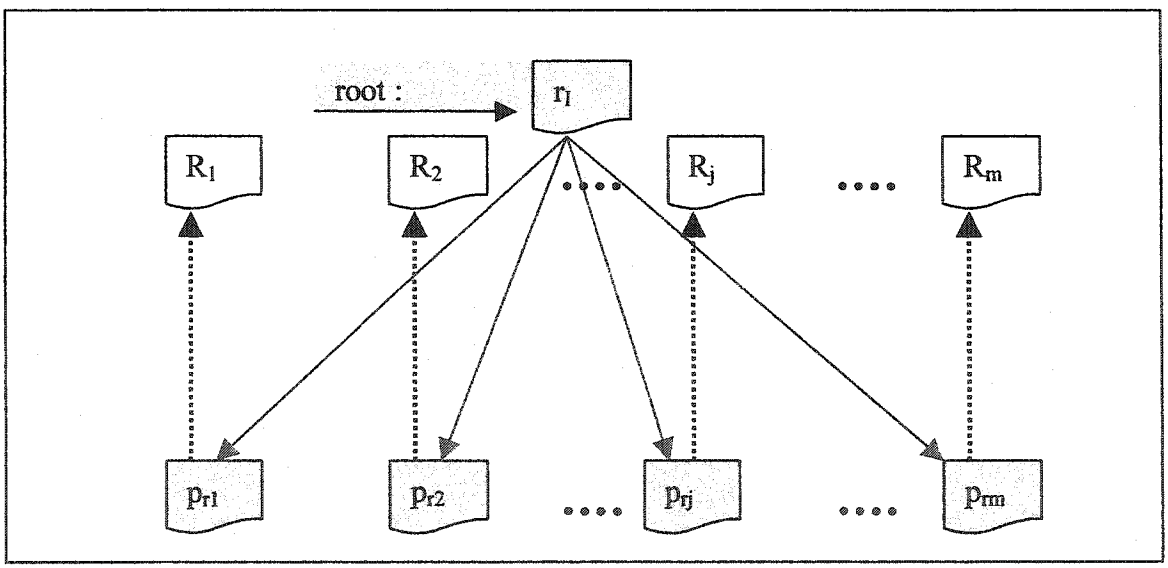

Figure 5. The sets of pages with incoming links from the pages in the candidate *request* partite set.

Step 6 Assign pages in the union of the m sets of R_i to the candidate *referer* partite set CR_F.

Step 7 Set N equal to the number of *referers* in the expanded CR_F.

NOTE:
Figure 6 contains sample matrices to illustrate the structure, labeling and content of the adjacency and hit frequency matrices generated by algorithm ***fBC*** to which matrix operations and heuristics are applied in Steps 8-24. These are provided as visual reference aids for the remainder of this section.

Steps 8-11 The *m-by-N* adjacency matrix $A_{rf \rightarrow rq}$ is instantiated to define all *referers* to request links between candidate partite members. Then the hit frequency of each edge *(rf, rq)*, link $rf \rightarrow rq$, is logged in hit frequency matrix $H_{rf \rightarrow rq}$. Given that the hit frequency $\omega(rf_f, rq_q) = h$, this means that there is a total of h entries in the access log where page rf_f refers to *request* page rq_q. The value h is assigned to row **q** column **f** of the matrix. Position-wise in the matrix, *requests* are the row headers while *referers* are the column headers.

Steps 12-22 * Preprocessing criteria of the candidate partite sets is designed to minimize the probability of less relevant pages moderating the make-up of the partite sets of the biclique neighborhood is now applied. Using hit frequency and page adjacency criteria, the m-by-N adjacency matrix $A_{rf \rightarrow rq}$ $CRQ \leftarrow CRF$ is used initially to log the hit frequency data. Hit frequency $\omega(\mathbf{rq},\mathbf{rf}) = k$ is the number of Web log entries in which *request* **rq** is referenced by *referer* **rf**. A direct link between two pages defines edge (**rf**,**rq**) for which adjacency is indicated by a $k>0$ entry in the matrix. Furthermore, edge (**rf**,**rq**) means that there exists at least one log entry in which page **rf** refers to *request* **rq**.

* The preprocessing criteria requires the (1) elimination of *referers* referencing less than 45% of the *request* candidates, and (2) elimination

of *requests* in which minimal interest is detected. The metric **Minimal Interest Level** ***(MinIL)*** of a *request* **rq** is defined in Illustration 5. The hit frequencies in the matrix are now replaced with '1' if $\mathbf{k \geq 1}$ and '0' if $\mathbf{k=0}$. A '1' entry in the adjacency matrix indicates that the pages are directly linked ($d = 1$), and a '0' entry indicates that no direct link exists. It is expected that a reduced and less sparse adjacency matrix of candidate partite set members will result from application of the preprocessing criteria. ***CRF*** and ***CRQ*** represent the modified candidate *referer* and *request* partite sets respectively.

Illustration 5. Minimal Interest Level metric used to rate page relevance.

```
METHOD minIL[K1, K2, medhf, rq]  /* Minimal Interest Level Detector */

   /* ω ≡ hit frequency
      medhf= Median{ω(rf, rq)} ≡ Median hit frequency of a request page rq.
      rf ≡ referer page;  rq ≡ request page
      K1 ≡ #(referers in CRF that refer to page rq),
      K2 ≡ #(referers in CRF).                                   */

      If  ( K1/K2 * Median{ω(rf, rq)} ≥ Median{ω(rf, rq) } )

            MinIL[rq] = 1;  /* interest detected */

         else

            MinIL[rq] = 0;  /*no interest detected */

             eliminate[rq]; }

      End_If.

END METHOD-MinIL
```

Steps 23-24 The adjacency matrix $A_{Q\leftarrow F} \equiv CR_Q \leftarrow CR_F$ identifies all candidate *referer* partite set to candidate *request* partite set page links **rf→rq** of length d=1. This matrix is derived from the preprocessed hit frequency matrix $H_{Q\leftarrow F}$ (produced in steps 8 to 18) simply by replacing the hit frequency entry of $\omega(rf,rq)_{qf} = k$ with a '1' if k > 0 and a '0' otherwise. The bi-directional adjacency matrix $A_{Q\leftrightarrow F}$ is preprocessed to remove from CR_F all pages *rf* referring to less than 45% of the set of *request* pages *rq* belonging to CR_Q. The set of pages not eliminated make up the *referer* partite set R_F.

Hit frequency (ω) matrix:

Edge: (rf,rq); ω(rf,rq) = *k*

$H_{Q\leftarrow F}$

rf→rq	rf1	rf2	rf3
rq1	3	6	0
rq2	1	4	15
rq3	5	5	8

Edge : (rf,rq); d(rf,rq) = 1

$A_{F\leftarrow Q}$

rf→rq	rf1	rf2	rf3
q1	1	1	0
rq2	1	1	1
rq3	1	1	0

Adjacency matrices:

Edge: (rq,rf); d(rq,rf)=1

$A_{Q\leftarrow F}$

rq→rf	rq1	rq2	rq3
rf1	1	1	1
rf2	0	1	1
rf3	1	1	0

Edges: (rf,rq), (rq,rf)
d(rf,rq) =1 and d(rq,rf) =1

$A_{Q\leftrightarrow F}$

rf↔rq	rf1	rf2	rf3
rq1	0	0	-1
rq2	1	0	0
rq3	0	0	0*

Figure 6. Structure, labeling and content of adjacency and hit frequency matrices.

Steps 27-31 Two maximal bicliques can be extracted by using heuristics on adjacency matrix $\boldsymbol{R}_{Q\leftrightarrow F}$. The strategy termed 'maximizing the *request* partite set' eliminates from $\boldsymbol{CR}_Q$ all *requests* with at least one non-zero or 0* row entry. The resulting biclique has *referer* partite set $\boldsymbol{R}_F = \boldsymbol{CR}_F$ and *request* partite set $\boldsymbol{R}_Q = \boldsymbol{CR}_Q - \boldsymbol{ZR}_Q$. Similarly, 'maximizing the *request* partite set' eliminates from $\mathbf{CR}_F$ all *referers* with at least one non-zero or 0* column entry. The resulting biclique has *request* partite set $\boldsymbol{R}_Q = \boldsymbol{CR}_Q$ and *referer* partite set $\boldsymbol{R}_F = \boldsymbol{CR}_F - \boldsymbol{ZR}_F$. The graphical representation of the biclique $\mathbf{B}_\lambda[(\boldsymbol{R}_F \cup \boldsymbol{R}_Q), \mathbf{E}_{f\leftrightarrow q}]$ generated by algorithm ***fBC*** is displayed in Figure 7.

END WALK-THROUGH of Algorithm fBC

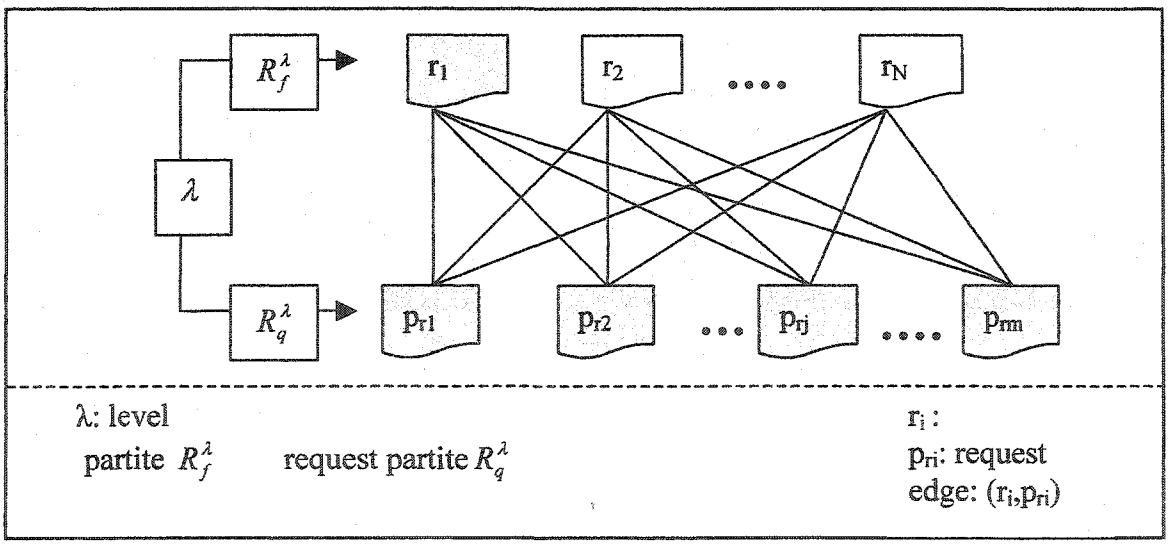

Figure 7. Biclique graph generated by algorithm *findBipartiteClique*(fBC).

The biclique infrastructure used to model the virtual search neighborhood sanctions the removal of edges so that no adjacency exists between any two pages that belong to the same partite set. This means that edge $(p_{ri}, p_{rj}) \notin E$ and edge $(r_i, r_k) \notin E$ of the virtual biclique graph $B_v(P,E)$. Pages in the same partite set have a similar status or relevance. Each page in the *referer* partite set has an outgoing link to each and every page in the *request* partite set. These direct links (d =1) define adjacency between two pages. Recall that research supports that the adjacency of two pages indicates topical and/or contextual relatedness and that hit frequency of a page is traditionally used to determine its relevance or value [8, 21]. The Web user's navigational search patterns of breath-first, depth-first, and/or a combination of the two are easily accommodated since links between adjacent pages can be traversed bi-directionally. Any pages affected by the removal of an edge to ensure the structural integrity of the biclique neighborhood are re-evaluated for membership in the neighborhood. The initial value of a page in the role of *referer* or *request* is established early in the preprocessing phase in which pages of high quality are identified and filtered through into distinct roles for populating the biclique search neighborhood.

3.3.2 Iterative Use of *fBC*

The algorithm used iteratively facilitates expanding the level λ=1 biclique search neighborhood into a lattice of hierarchally generated bicliques. With the addition of a choice structure at the beginning of algorithm ***fBC***, the remainder of the algorithm requires only minor adjustments. The essential modifications to the algorithm that facilitate the iterative use of ***fBC*** are provided in Illustration 6.

Illustration 6. Modification of algorithm *fBC* required to implement its iterative use.

If $\lambda = 1$ **then**

1 *Input* **initial** r_i //* *Web user selected page, URL ε Website Ws (C_hLF access log structure* *//

2 *Assign* r_i **to partite** R_f //* *biclique characterized as rooted in* r_i; $Min(\#_{R_f})=1$ *//

3 *Determine* $CRQ = \{$ **all pages with** $r_i\}$ //* *Candidate request partite set* *//

else /* **i.e.** $\lambda > 1$ */

/* ***No transformations of*** $RF^{\lambda+1}$ ***or*** $CRF^{\lambda+1}$ ***are allowed after instantiation of actual partite set at level*** $\lambda > 1$ */

1* **Set** $RF^{\lambda+1} = RQ^{\lambda}$ /* *Switch role of level* λ *request partiteset to level* $(\lambda+1)$ *actual partite set.* */

2a* **Set** $CRF^{\lambda+1} = RF^{\lambda+1}$ /*

2b* **Set** $m = \#(RF^{\lambda+1})$

3* **Choose** $r_i \varepsilon RF^{\lambda+1}$ /* Web user *preference facilitated at level* $\lambda+1$ */

/* ***Initial assignment of pages to actual request partite*** $RQ^{\lambda+1}$***. Request pages used in any previously generated biclique*** B_λ ***are removed from inclusion at the current level.*** */

4* **Assign** $CRQ^{\lambda+1} = \{p_j \mid r_i \rightarrow p_j$ **and** $p_j \notin CRQ^k, k < \lambda+1\}$

/* ***Expand level (k >1) candidate request partite set*** $CRQ^{\lambda+1}$ */

5* *For* **each page** $p_j \varepsilon RF^{\lambda+1}$ $(1 \leq j \leq m)$ /* *repeat* 5.1 *m times* */

/* ***Determine*** R_j***, set of pages with out-links from*** $p_j \in RF^{\lambda+1}$ ***not used at previous levels*** */

*5.1** $R_j \equiv \{p_j \mid p_j^k \rightarrow p_i^{(k+1)}$ **and** $p_i \notin CRQ^k$ **for** $k < \lambda+1\}$

6* $CRQ^{\lambda+1} = \bigcap_{1 \leq j \leq m} R_j$ /*candidate request partite set defined */

7* **Set** $N = \#(CRQ^{\lambda+1})$

The use of this strategy as pictorially illustrated in Figure 8 requires that the role of the current *request* partite set be switched to that of *referer* at each successive level λ. Therefore, the *request* partite page set $\{p_{r1}, p_{r2}, \ldots, p_{rm}\}$ at level $\lambda = L$ becomes the actual

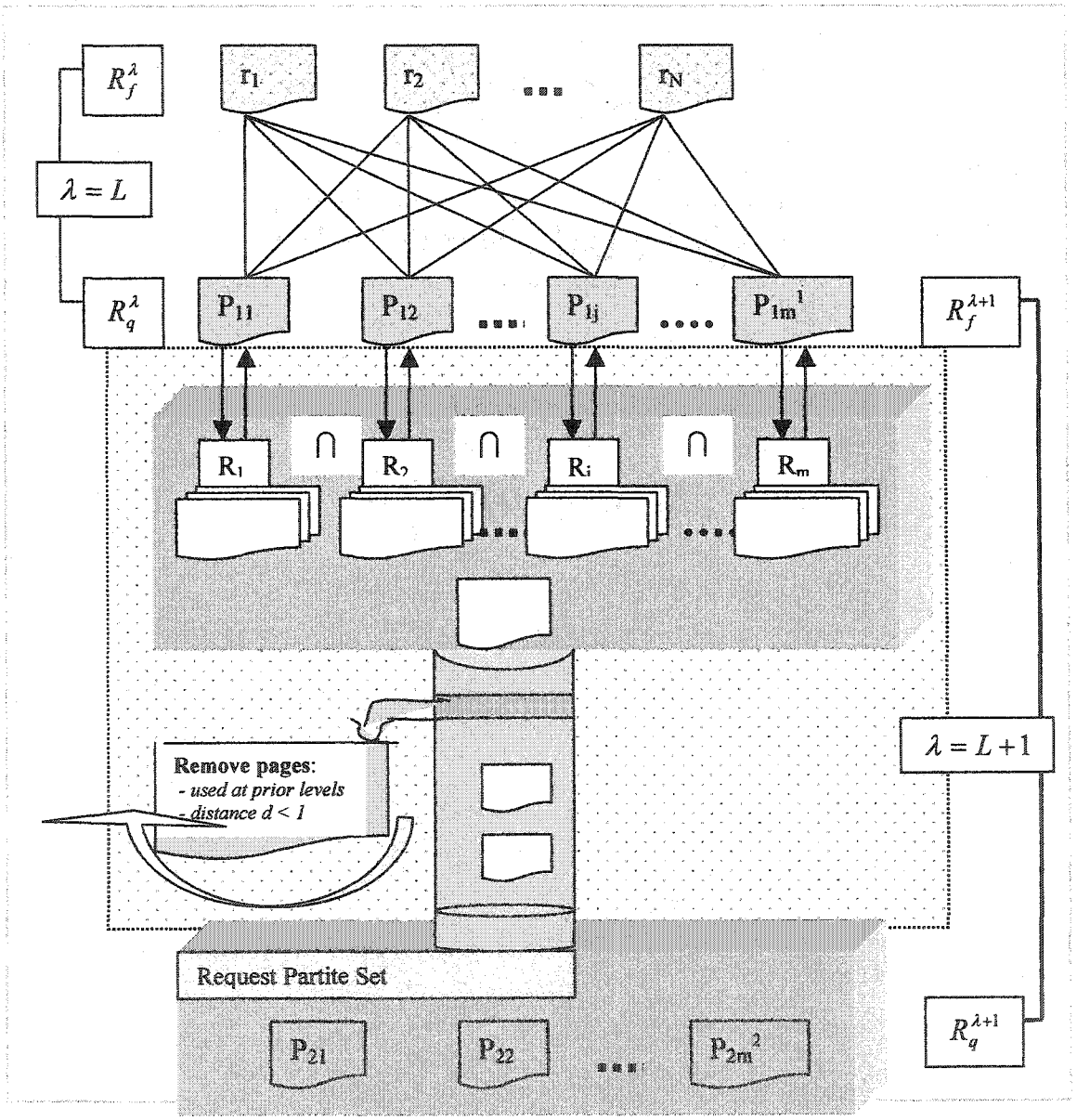

Figure 8. Iterative use of *fBC* employs current *request* partite set as an articulation hub of *referers* to generate the next level biclique.

referer partite set $\mathbf{R}_F^{\lambda+1}$ of the level λ=L+1 generated biclique. Hence, no transformation of $\mathbf{R}_F^{\lambda+1}$ is allowed after its initial instantiation at level λ+1.

The requisite user preference, as represented by his choice of a new $\mathbf{r}_i$ at each succeeding level of the expanded neighborhood infrastructure, is reinforced by requiring the level λ=L+1 *referer* partite set to be the articulation hub for the generation of the level λ=L+1 *request* partite set. Step 1 of the modified ***fBC*** algorithm requires the user to supply the root page $\mathbf{r}_i$ (or object) of interest at the beginning of each successive iteration. However, for level λ = c>1, the user's selection of r_i is restricted to pages that belong to the level λ=c *referer* partite set (formerly level λ = c-1 *request* partite set). The candidate request partite set $\mathbf{R}_Q^{\lambda}$ is assigned all request pages $\mathbf{p}_i$ for which the user's page of choice $\mathbf{r}_i$ is the *referer*. The user's preference is thus reinforced algorithmically by his choice of $\mathbf{r}_i$ at the start of each subsequent level of the dynamically structured biclique-lattice.

Using ***fBC*** iteratively, each *request* partite set $\mathbf{R}_Q$ becomes an *articulation hub* between the currently generated biclique subgraph rooted in user preference (as represented by page choice $\mathbf{r}_1$) and any other successively generated biclique subgraphs of the lattice. To generate successive biclique subgraphs rooted in the user preference, the level-λ *request* partite set becomes the level-(λ+1) actual *referer* partite set. The intersection of the sets of *request* pages identified per level-(λ+1) *referer* set is used to identify the candidate *request* partite page set for this level. Figure 9 demonstrates graphically the hierarchal organization of the biclique-lattice generated via the iterative use of ***fBC*** algorithm.

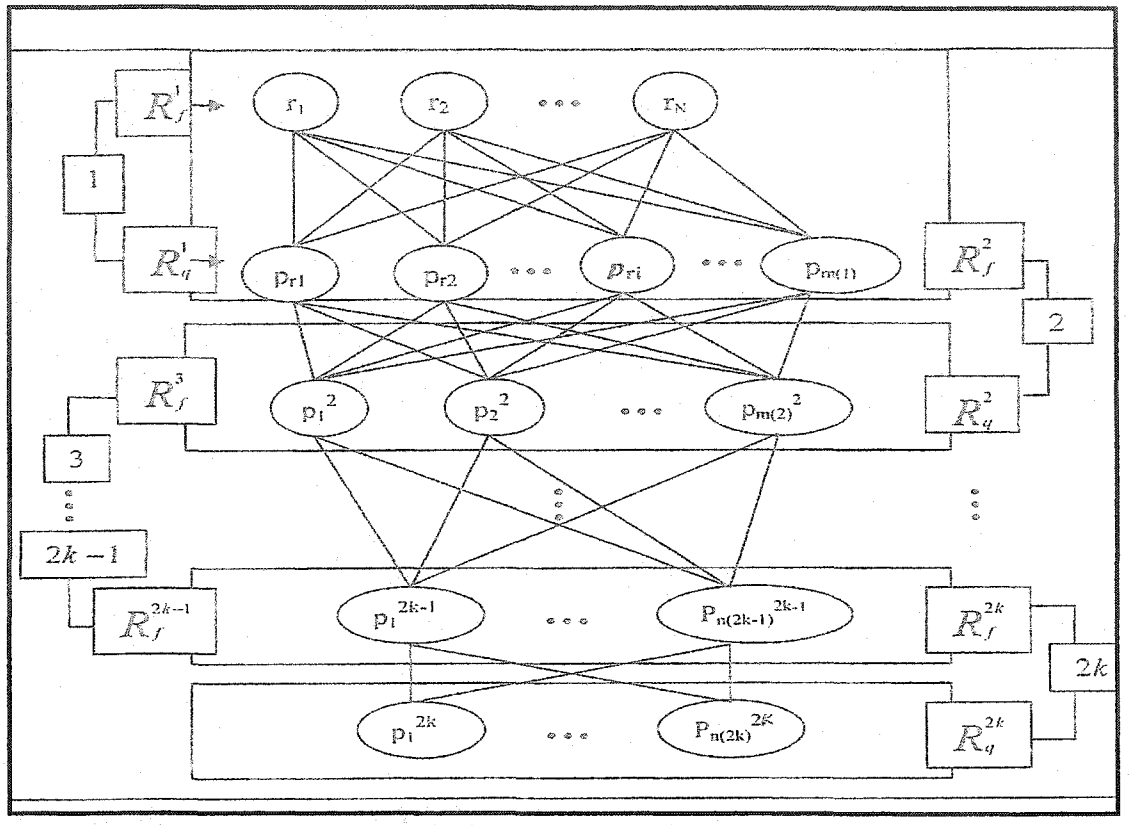

Figure 9. Biclique-lattice generated by the iterative use of *fBC*.

3.4 Experiment

Experimental results produced by algorithm *fBC* using data set *D* extracted from Music Machine Archive's C_bLF Web server logs follows.

3.4.1 *fBC* Generates Biclique Search Neighborhood

Step 1. Input: initial search *URL* $\mathbf{r_I}$ = http://www.hyperreal.com/music/machines/.

Observation(s):

(1) This URL is the Web user's initial choice that signifies his search interest.

Step 2. Assign $\mathbf{r_I}$ = http://www.hyperreal.com/music/machines/ to the *referer* partite set.

Observation(s):

(1) This is the first page assigned to the actual *referer* partite set of the biclique neighborhood. Hence, the search is now characterized as rooted in $\mathbf{r_I}$, or rooted in the user's preference.

(2) This assignment guarantees that user preference will moderate the selection of pages used to populate the biclique neighborhood.

(3) This also guarantees that the *referer* partite set will have at least one page.

Step 3. The 11 pages, rq_{NO}: 1, 3, 4, 5, 7, 8, 9, 12, 13, 20, 21, of the candidate *request* partite set $\mathbf{CR_Q}$ are identified in Table 2.

Observation(s):

(1) User selected page $\mathbf{r_I}$ has twenty-six (26) out-links. This means that it refers to 26 distinct *requests* **rq**.

(2) *Request* pages at a distance $\mathbf{d \neq 1}$ from *referer* $\mathbf{r_I}$ are excluded because they do not meet the adjacency criteria.

(3) Edge length ∞ means that r_1 refers to a page that belongs to another Website.

(4) Hit frequencies of pages referenced by r_1 range from 1,089 down to one.

Table 2. Candidate *request* set pages identified by *fBC* and their hit frequency.

Web user's initial page choice: r_1		out- links: #26	
$rq_{No.}$	ω	*request* (rq_i)	$d(r_1,rq_i)$
1	1089	» U:/music/machines/samples.html	1
2	1058	U:/music/machines/	0
3	838	» U:/music/machines/manufacturers/	1
4	694	» U:/music/machines/guide/	1
5	448	» U:/music/machines/images.html	1
6	381	U:/music/machines/manufacturers/Roland/TB-303/	3
7	320	» U:/music/machines/links/	1
8	276	» U:/music/machines/search.html	1
9	257	» U:/music/machines/new.html	1
11	199	U:/music/machines/manufacturers/Moog/	2
12	134	» U:/music/machines/ecards/	1
13	77	» U:/music/machines/index.html	1
14	148	U:/music/machines/manufacturers/Roland/MC-303/	3
15	90	U:/music/machines/adaptive/browsing.html	2
16	119	U:/music/machines/manufacturers/Moog/Modular/	3
17	58	U:/music/machines/manufacturers/ARP/	2
18	46	U:/music/machines/adaptive/notfound.html	2
19	43	U:/music/machines/adaptive/found.html	2
20	9	» U:/music/machines/email.html	1
21	2	» U:/music/machines/Analogue-Heaven/	1
22	2	U:/wc/-d/6/-c/21/-r/-z/musicmachines	∞
23	1	U:/machines/images/ftrmap.map?430,31	∞
24	1	U:/raves/	∞
25	1	U:/raves/grid/	∞
26	1	U:/raves/holdyourown/	∞

Step 4. $m = 11$

Observation(s):

(1) ***m***, the number of pages initially assigned to the candidate *request* partite set.

Steps 5-6. The set of *referers* (**CRF**) to each of the 11 respective *requests* in **CRQ** that meet the adjacency requirement $\mathbf{d(r_j, rq_i) = 1}$ are charted as the 19 row-headers in the adjacency matrix A represented in Table 3.

Observation(s):

(1) Table 3, sectioned into four parts, represents the 12-by-19 adjacency matrix A. The *requests* are the row-headers and the *referers* the column-headers.

(2) 19 *referers* are found in the data set that meet the adjacency criteria.

(3) The actual partite set members will come from these candidate partite sets.

Step 9 Application of preprocessing criteria to adjacency matrix **A** eliminates candidate *referer* set pages r_5, r_8, r_{10}, r_{11}, r_{12}, r_{15}, r_{16}, r_{18}, and r_{19}.

Observation(s):

(1) The reduced candidate *referer* set pages are r_1, r_2, r_3 , r_4, r_6, r_7.

(2) 49% of the pages in the candidate *request* partite set are eliminated.

Table 3. $A_{rf \to rq}$ **:*12x19 referer-to-request* adjacency matrix of candidate partite sets.**

	rf / *rq*	(root) http: r_1	guide r_2	links r_3	index. html r_4	samples. html r_6	manufac-turers r_7
rq0	*U:r1*	---	1	1	1	1	1
rq1	*samples.html*	1	0	1	1	---	1
rq2	*manufacturers*	1	1	1	1	1	---
rq3	*guide*	1	---	1	1	1	1
rq4	*images.html*	1	0	0	1	0	1
rq5	*links*	1	1	---	1	1	1
rq6	*search.html*	1	1	1	1	1	1
rq7	*new.html*	1	0	0	1	0	0
rq8	*ecards*	1	0	0	1	0	0
rq9	*index.html*	1	1	1	---	1	1
rq10	*email.html*	1	1	0	0	0	0
rq11	*Analogue-Heaven*	1	0	0	0	0	0

Table 3. (continued)

	rf / *rq*	adap-tive r_5	images. html r_8	catego-ries r_9	prices r_{10}	new. html r_{11}	schema-tic. html r_{12}
rq0	*U:r1*	0	0	0	0	0	0
rq1	*samples.html*	0	0	1	0	0	0
rq2	*manufacturers*	1	1	1	1	1	1
rq3	*guide*	0	0	0	0	1	0
rq4	*images.html*	0	---	1	0	0	0
rq5	*links*	0	1	1	1	1	0
rq6	*search.html*	0	1	1	1	1	1
rq7	*new.html*	0	0	0	0	---	0
rq8	*ecards*	0	0	0	0	1	0
rq9	*index.html*	1	1	1	0	1	1
rq10	*email.html*	0	0	0	0	0	0
rq11	*Analogue-Heaven*	0	0	0	0	0	0

Table 3. (continued)

	referrer / *request*	search. html r_{13}	Software. html r_{14}	Analogue-Heaven r_{15}	ecards r_{16}
rq0	*U:r1*	0	0	1	1
rq1	*samples.html*	0	0	0	0
rq2	*manufacturers*	1	1	0	0
rq3	*guide*	1	1	0	0
rq4	*images.html*	0	0	0	0
rq5	*links*	1	1	0	0
rq6	*search.html*	---	1	0	0
rq7	*new.html*	0	0	0	0
rq8	*ecards*	0	0	0	---
rq9	*index.html*	1	1	0	0
rq10	*email.html*	1	0	0	0
rq11	*Analogue-Heaven*	1	0	---	0

Table 3. (continued)

	rf / *rq*	email.html r_{17}	gearlists/ r_{18}	mods. html r_{19}
rq0	*U:r1*	0	0	0
rq1	*samples.html*	0	0	0
rq2	*manufacturers*	1	1	0
rq3	*guide*	1	0	0
rq4	*images.html*	0	0	0
rq5	*links*	1	1	1
rq6	*search.html*	1	0	0
rq7	*new.html*	0	0	0
rq8	*ecards*	0	0	0
rq9	*index.html*	1	1	1
rq10	*email.html*	---	0	0
rq11	*Analogue-Heaven*	0	0	0

Step 10. Calculate the hit frequency matrix $\mathbf{H}_{rf \rightarrow rq}$ on reduced candidate partite sets resulting from Step 6. See Table 4.

Table 4. Hit frequency matrix using reduced candidate partite sets of preprocessed adjacency matrix A.

rf / rq		(root) http: r_1	guide r_2	links r_3	index. html r_4	samples. html r_6	manufac-turers r_7
rq0	*U:r1*	(1058)	1	2	13	2	5
rq1	*samples.html*	1089	0	1	69	(2)	95
rq2	*manufacturers*	838	153	7	134	13	(14)
rq3	*guide*	694	(161)	3	63	2	2
rq4	*images.html*	448	0	0	38	0	31
rq5	*links*	320	36	(11)	43	8	39
rq6	*search.html*	276	96	10	28	13	43
rq7	*new.html*	257	0	0	34	0	0
rq8	*ecards*	134	0	0	13	0	0
rq9	*index.html*	77	21	4	(15)	10	12
rq10	*email.html*	9	5	0	0	0	0
rq11	*Analogue-Heaven*	2	0	0	0	0	0

Table 4 (continued)

rf / rq		catego-ries r_9	new. html r_{11}	search. html r_{13}	Software. html r_{14}	email. html r_{17}
rq0	*U:r1*	0	0	0	0	0
rq1	*samples.html*	8	0	0	0	0
rq2	*manufacturers*	14	31	72	6	1
rq3	*guide*	0	4	31	2	3
rq4	*images.html*	8	0	0	0	0
rq5	*links*	7	47	40	3	8
rq6	*search.html*	8	12	(10)	2	1
rq7	*new.html*	0	---	0	0	0
rq8	*ecards*	0	9	0	0	0
rq9	*index.html*	2	5	14	5	3
rq10	*email.html*	0	0	1	0	(3)
rq11	*Analogue-Heaven*	0	0	56	0	0

Steps 12-22. Application of the ***MinIL*** metric to the candidate *requests* partite set (row-headers) used in the hit frequency Table 4 eliminates pages *new.html, ecards, email.html and Analogue-Heaven.*

Observation(s):

(1) 33% of the pages in the candidate *request* partite set are determined to have no current relevance.

(2) The **MinIL** metric detects current page relevance for page index.html and no current page relevance for *request* pages new.html, even though the 77 hit frequency of page index.html is lower than the 256 hit frequency of page new.html. Consider the following calculations of **MinIL** (index.html) and **MinIL** (new.html).

(a) The median hit frequency value of the candidate *request* partite set is 293.

(b) **MinIL**: $K_1 * 20\% * medhf \geq medhf$. For page index.html (10*0.2*293) = 586 > 293, so that **MinIL**(index.html)=1. Therefore, page index.html is not eliminated because k=10 means that 10 authoritative pages (*referers* in **CRF**) identify it as having thematic relevance to the user's initial page choice.

(c) **MinIL**: $K_1 * 20\% * medhf \geq medhf$. For new.html (1 * 0.2 * 293) = 58.6 < 293, so that **MinIL** (new.html) = 0. Therefore, new.html is eliminated because it is referenced by only one *referer* in **CRF**.

Steps 23. Instantiate the 6-by-11 adjacency matrix after application of the ***MinIL*** detector on the candidate *request* partite set.

Table 5. Adjacency matrix to establish *referer* to *request* page links of distance 1.

rf → rq	(root) r_1	guide r_2	links r_3	Index.html r_4	samples.html r_6	manufacturers r_7	Categories r_9	new.html r_{11}	Search.html r_{13}	software.html r_{14}	email.html r_{17}
rq1	1	0	1	1	---	1	1	0	0	0	0
rq2	1	1	1	1	1	----	1	1	1	1	1
rq3	1	---	1	1	1	1	0	1	1	1	1
rq5	1	1	----	1	1	1	1	1	1	1	1
rq6	1	1	1	1	0	1	1	1	---	1	1
rq9	1	1	1	----	1	1	1	1	1	1	1

Steps 23. Adjacency matrix to establish the *request* to *referer* page links between the candidate partite sets.

Table 6. Adjacency matrix B to establish the *request* to *referer* links of distance 1.

rq → rf	(root) r_1	guide r_2	links r_3	Index.html r_4	samples.html r_6	manufacturers r_7	Categories r_9	new.html r_{11}	Search.html r_{13}	software.html r_{14}	email.html r_{17}
rq1	1	0	1	1	---	1	1	0	0	0	0
rq2	1	1	1	1	1	----	1	1	1	1	1
rq3	1	---	1	1	1	1	0	1	1	1	1
rq5	1	1	----	1	1	1	1	1	1	1	1
rq6	1	1	1	1	0	1	1	1	---	1	1
rq9	1	1	1	----	1	1	1	1	1	1	1

Observation(s):

(1) Table 6 represents the transpose of the11x6 matrix produced by the procedure described in Step 33.

(2) Note: B[2][17]=1 means that link $rq_2 \rightarrow r_3$ exists, i.e., there is a direct link of length or distance 1 from *request* page rq_2 to page r_3.

Steps 24-26. Matrix operations used to calculate the bidirectional adjacency matrix C in Table 7.

Table 7. The 6x11 adjacency matrix C representing the *request-referer* candidate partite sets' bi-directional page adjacencies rf↔ rq.

rf ↔ rq	(root) r_1	guide r_2	links r_3	Index. html r_4	samples. html r_6	manufac-turers r_7	Catego-ries r_9	new. html r_{11}	Search. html r_{13}	software. html r_{14}	email. html r_{17}
rq1	1	0	1	1	---	1	0	0	0	0	0
rq2	1	1	1	1	1	----	1	0	1	1	0
rq3	1	---	1	1	1	1	0	0	1	0	1
rq5	1	1	----	1	1	1	0	0	1	0	0
rq6	0	1	1	1	0	1	0	0	---	0	0
rq9	1	1	1	---	1	1	0	1	1	0	0

Steps 27-32. Heuristics are used on linkage data in Table 8 to extract two maximal biclique neighborhoods.

Table 8. Alpha-adjacency matrix for the candidate *referer–request* partite sets.

rf ↔ rq	(root) r_1	guide r_2	links r_3	Index. html r_4	samples. html r_6	manufac-turers r_7	Search. html r_{13}
rq1	1	0	1	1	---	1	0
rq2	1	1	1	1	1	----	1
rq3	1	---	1	1	1	1	1
rq5	1	1	----	1	1	1	1
rq6	(0)	1	1	1	0	1	---
rq9	1	1	1	----	1	1	1

A. Maximize the *referer* partite set $\mathbf{R}_F$. The resulting biclique has *referer* partite set $\mathbf{R}_F$ and *request* partite set $\mathbf{R}_Q$ defined as follows:

$\mathbf{R}_F$ = {r1, r2, r3, r4, r5, r6, r7, r13}

$\mathbf{R}_Q$ = {rq2, rq3, rq5, rq9}

B. Maximize the *request* partite set $\mathbf{R}_Q$. The resulting biclique has partite set $\mathbf{R}_F$ and *request* partite set $\mathbf{R}_Q$ defined as follows:

$\mathbf{R}_F$ = {r1, r3, r4, r6, r7}

$\mathbf{R}_Q$ = {rq1, rq2, rq3, rq5, rq9}

Observation(s):

(1) *Request* rq_6 is eliminated from the *request* partite set because page r_1 is the gate-keeper of the biclique neighborhood. Each *request* page must have an outgoing link with its terminus in r_1, the user's initial page choice.

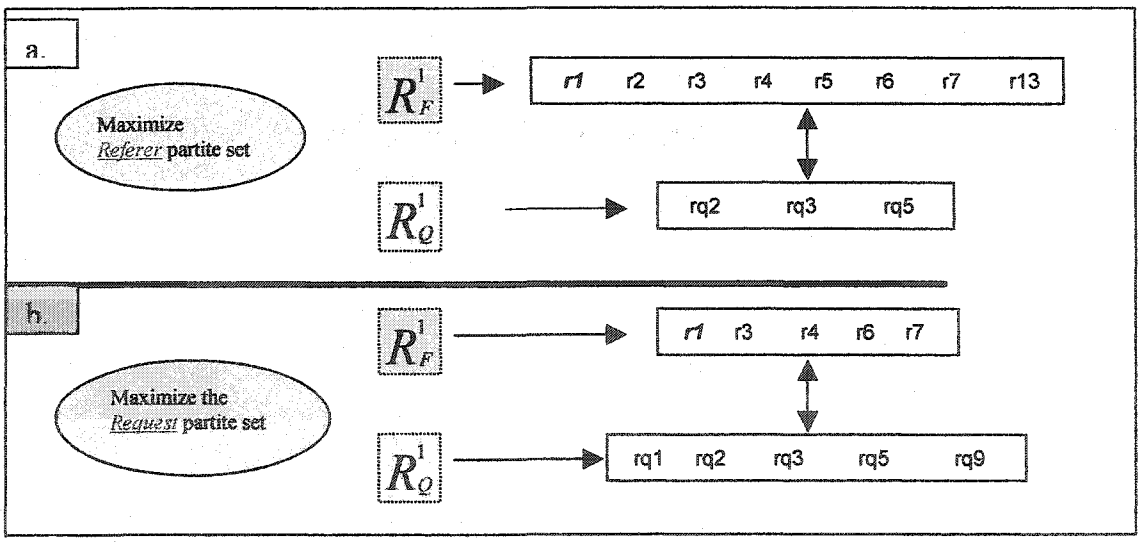

Figure 10. Snapshot of the two maximal bicliques generated by *fBC* and rooted in user preference r_1.

3.4.2 Expansion of Biclique Search Neighborhood

The iterative use of algorithm ***fBC*** (Illustration 6) starts with Step 1* of the *else* option of the algorithm's choice structure. The biclique $\mathbf{B_B}$ in Figure 10b describes the algorithmic modification used from this point in the experiment:

Step 1.* The level $\lambda = 1$ *request* partite set $\mathbf{R}Q^1$ takes on the role of level $\lambda = 2$ *referer* partite set $\mathbf{R}F^2$. (Refer back to the biclique-lattice in Figure 9). Therefore, at level $\lambda = 2$ the specific pages assigned to the *referer* partite set $\mathbf{R}F^2 \equiv \mathbf{R}Q^1 =$ ***{manufacturers, guide, links, index.html,samples.html}*** is predetermined.

Step 2a.* Set the candidate *referer* partite set to actual *referer* partite set, i.e., $\mathbf{C}\mathbf{R}F^2 = \mathbf{R}F^2$.

Observation(s):

(1) The label of a 'candidate *referer* partite set' at level $\lambda > 1$ is for algorithmic convenience associated with the naming and referencing of data structures since the set of pages belonging to the actual *referer* partite set is fixed in Step1.*

Step 2b.* $m = \#(\mathbf{R}F^2) = 4$

Step 3.* Input $\mathbf{r_i} \in \mathbf{R}F^2$: $\mathbf{p_i}^2 \equiv$ *http:*$\mathbf{r_1}$ = **http:/.../music/machine/manufacturers/.**

Observation(s):

(1) The user's selection is restricted to one of the five pages in the *referer* partite set. Note that this restriction applies at level $\lambda > 1$.

(2) p_i^2: page i at level $\lambda = 2$.

Step 4.* Level $\lambda = 2$ candidate *request* partite set $\mathbf{C}\mathbf{R}Q^2$ is initialized with the assignment of all *request* pages $\mathbf{rq_i}$ with incoming links from $\mathbf{r_1} \equiv \mathbf{p_1}^2$ such that

$d(p_1^2, rq_i) = 1$ and rq_i is not a page used at level $\lambda = 1$. Table 9 gives the list of all pages for which r_1 is *referer* along with their respective hit frequencies. The candidate *request* partite set is CRQ^2={**categories, schematics.html, gearlists, software, prices, mods.html, images.html**} initially.

Table 9. List of pages (URL) referenced by http:/.../music/machine/manufacturers/.

Data set : Music Machine Access Log

: $p_i^2 \equiv r_1$= **http:/.../music/machine/manufacturers/** ***out- links*: 11**

$d = 1$

$rq_{No.}$	ω	R_1:*request* (rq_i)
1	276	» U:/music/machines/categories/
2	198	» U:/music/machines/schematics.html
3	157	» U:/music/machines/gearlists/
4	119	» U:/music/machines/software.html
5	95	U:/music/machines/samples.html
6	69	» U:/music/machines/prices/
7	43	U:/music/machines/search.html
8	41	» U:/music/machines/mods.html/
9	31	» U:/music/machines/images.html
10	14	*U:/music/machines/manufacturers/
11	12	U:/music/machines/index.html
12	2	U:/music/machines/guide/

*rq_{10}: $d = 0$

» pages that do not belong to any prior partite levels

*Step 5.** The set R_j of *request* pages pointed to by *referer* http:$p_j \in R_F^2$ $(1 \leq j \leq 5)$ is listed in column *j* of Table 10. The table's column headers name the five *referers* in R_F.[2] The set R_j of requests is enumerated as column entries under their respective headers.

Table 10. The set of *requests* referenced by each *referer* page p_j at level λ=2.

http: $p_j \rightarrow$	manufacturers	guide	index.html	link
URL: rq_j	$\downarrow R_1$	$\downarrow R_2$	$\downarrow R_3$	$\downarrow R_4$
	search.html	search.html	search.html	search.html
	schematics.html		new.html	addressbook
	software.html		images.html	
R_j	prices		ecards	
	mods.html			
	images.html			
	gearlist.html			
	categories			

*Step 6.** The intersection of the five R_j *s* defines the candidate request set at level λ=2. Specifically, CR_Q^2 is the intersection $\cap_{1 \leq j \leq 5} R_j$ = {URL:/.../search.html }. Table 10 validates this find. Figure 11 graphically illustrates the 2-level biclique lattice generated by one iteration of *fBC*.

*Step 7.** Set N= $\#(CR_Q^2) = 1$

*Steps 8-26.** Any instructions or sequence of instruction of algorithm *fBC* that would transform the *referer* partite set R_F^2 are conditionally omitted for levels λ>1.

The notation used to represent the level $\lambda = 1$ and level $\lambda = 2$ bicliques is $B^1(R_F{}^1 \cup R_Q{}^1, F \leftrightarrow Q)$ and $B^2(R_F{}^2 \cup R_Q{}^2, F^2 \leftrightarrow Q^2)$ respectively. The notation for the biclique-lattice is $B\text{-}\mathcal{L}(B^1 \S B^2)$.

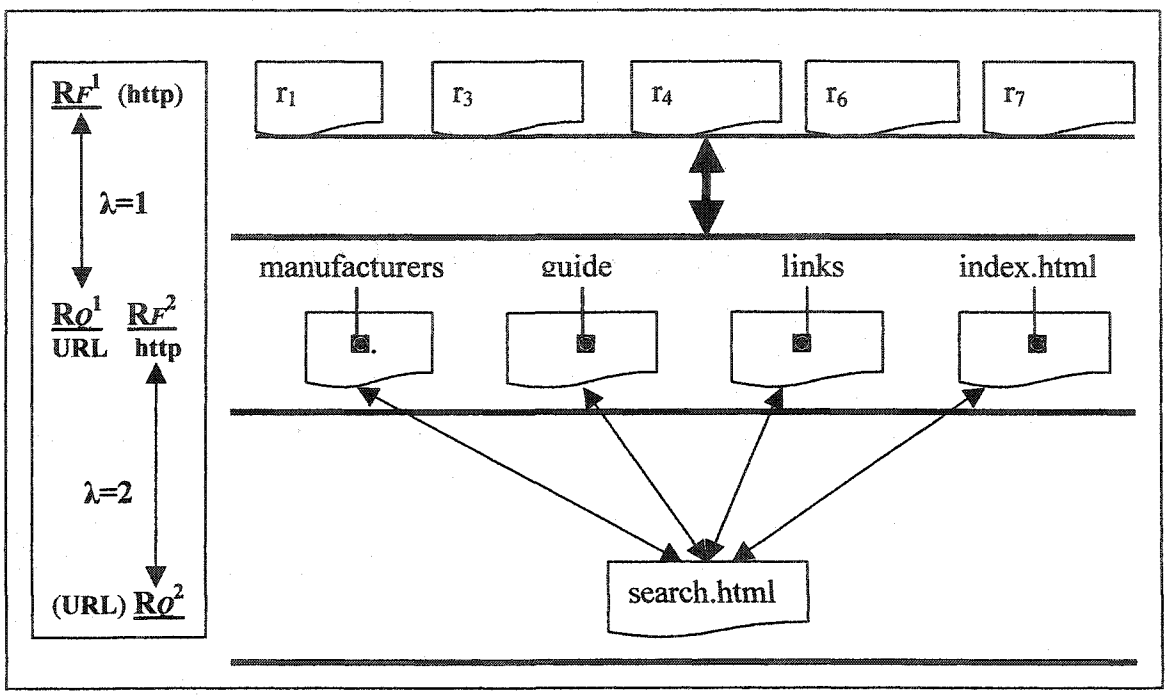

Figure 11. The level $\lambda = 2$ biclique-lattice generated using *fBC* iteratively.

Observation(s):

(1) The *request* partite set has one page. With 1140 distinct pages/objects in the test data set, the size of the level-2 biclique *request* partite set may be a reflections of bias in the selection pool of the experimental data set, or structural inadequacies of the Website. However, the algorithm *fBC* demonstrates it ability to extract bicliques from C_bLF access logs.

3.4.3 User Directed Restructuring

The level one biclique neighborhood generated by ***fBC*** is rooted in the user's preference. The Web user chooses the Web object **http:r_1** that initiates his search. Employing hub-authority dynamics, Web pages at the host Website that meet the criteria of spatial and thematic and/or contextual value to the user's search initiative are identified and assigned to either the *referer* or *request* partite sets. The user's search interest or preference is signified by his initial choice of page r_1. This initial page is made the first member of the *referer* partite set and acts as gate-keeper to the set.

By setting the level-1 biclique neighborhood as a portal to the search neighborhood customized to the Web user's specific search agenda, restructuring is a most effective strategy to facilitate Website responsiveness to the user's immediate and observable navigational intent [28]. The ***fBC*** algorithm is designed to first assess the user's immediate intent (as indicated by his choice of a page in which the biclique neighborhood is said to be rooted) and uses the collective user's history (Web access log data) to determine what pages of relevance are available to like-intentioned users who have arrived at a similar or same point in their search activity, and then use the add-link restructuring activity to create paths to any of these pages that support the user's search objective. This approach thus allows for expanding/extending the levels of the biclique-lattice neighborhood.

Website restructuring includes the primary structure modification activities of: (1) add/delete page, (2) relocate page, (3) modify page content, and (4)add/remove/ modify intra- and/or inter-page links. The biclique search infrastructure promoted in this research utilizes structure modification activity four to enforce the spatial and link criteria

defined by α-adjacency. Pages p_1 and p_2 are α-adjacent if and only if (1) p_1 and p_2 belong to different partite sets in the same biclique subgraph (i.e., $p_1 \in A_{rf}$-partite set and $p_2 \in A_{rq}$-partite set of biclique graph A), (2) p_1 and p_2 are linked bi-directionally (i.e., $p_1 \leftrightarrow p_2$ means that $p_1 \rightarrow p_2$ and $p_1 \leftarrow p_2$), and (3) the distance between p_1 and p_2 is one (i.e., $d(p_1,p_2) = 1$ and $d(p_2,p_1) = 1$).

The tri-paginal link is introduced in this research as the structural unit of the biclique-lattice generated via the iterative use of our algorithm ***fBC***. This structural metric gauges the spatial qualities of two pages relative to a third page called the nucleus (a *request* page) for α-adjacencies (Illustration 7). The ***3-pLM*** in Figure 12 is gauged specifically for the intended purpose of expanding the biclique-lattice neighborhood in Section 3.3.2

Illustration 7. The 3-pLM used to determine current relevance (fitness) of a request page to the user's search interest.

1. **Set** the candidate *request* page $\mathbf{rq_N}$ as the nucleus.
2. **Determine** if α-distance exists between the nucleus and the user's current page preference $\mathbf{r_j}$ **(i.e.,** $\mathbf{d_\alpha(r_j,rq_N)=1}$ or $\mathbf{d_\alpha(rq_N,r_j)=1}$ **).**
3. **IF** $d_\alpha=1$ (for either directional link)
4. **Determine** each distinct 3-pL with pages in the *referer* partite set serving as E-W nodes.
   Add links to create α-adjacencies between these pages.
   Assign *request* (nucleus) page to the *request* partite set.
5. **Else**
6. **Eliminate** $\mathbf{rq_N}$ from current candidate *request* partite set.
7. **End IF**

by means of the add-link restructuring activity directed toward its *request* partite set which contains only one relevant page. The tri-paginal link metric is used to provide a means of measuring a candidate request page's fitness.

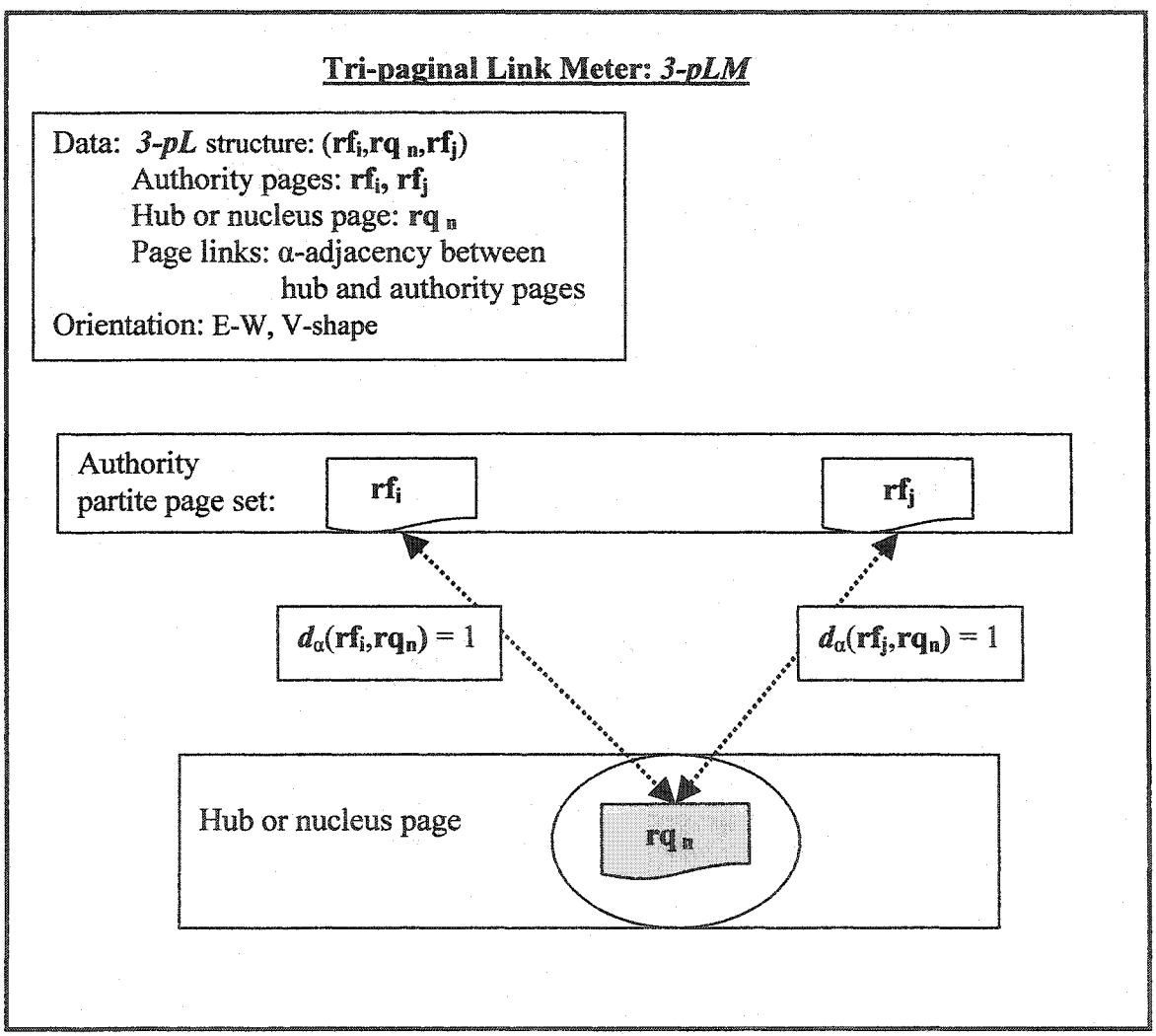

Figure 12. The *3-pLM* is set to ensure spatial and thematic relatedness of pages populating the biclique-lattice search neighborhood generated by *fBC*.

With the overriding objectives of using restructuring (particularly the addition of links) to expand/extend the biclique-lattice search neighborhood resulting from the experiment (see Figure 11) and preserving the thematic and spatial qualities of the extended search neighborhood, it is essential that algorithm ***fBC*** be made to ensure user preference at every level $\lambda>1$. Consideration of Figure 11 shows that the biclique generated at level $\lambda=2$ has only one page (**U:/music/machine/ search.html**) in its *request* partite set. The negation of user preference is obvious since the user's choice of page $\mathbf{r_j}$ = **'http:/music/machine/manufacturers,'** in which the level $\lambda=2$ biclique subgraph is rooted, identifies seven pages of relevance to the user's search (Table 10). Clearly, the user's choice is overridden by the collective user's access history and/or the structure of the Website. How? Either, Web users have not accessed *request* pages needed to identify page links that do exists (this may signify bias in the data set), or there are no links giving access to *requests* pages identified as relevant by the user's current page choice. Therefore, the intersection of the sets of *request* per *referer* will most likely not include the majority, if any, of the *requests* referenced by the user's page of choice $\mathbf{r_j}$. To rectify this negation of user preference, restructuring is employed.

Starting with **Step 3*** in the modification of algorithm ***fBC*** (see Illustration 4), the user's choice of a page of interest is restricted to the set of pages making up the actual *referer* partite set at level $\lambda>1$. **Step 4*** assigns to the candidate *request* partite set $\mathbf{CRQ^{\lambda+1}}$ all *request* pages for which the current user's page of choice $\mathbf{r_j}$ is *referer*. Now, to get around the impasse of negating user preference, **Step 4.1*** is incorporated to assign all pages in the level $\lambda>1$candidate *request* partite set $\mathbf{CRQ^{\lambda+1}}$ to the actual *referer* partite set $\mathbf{RQ^{\lambda+1}}$. This modification makes it mandatory that all pages referred to by the user's

page of choice $\mathbf{r_j}$ be constituted as pages of relevance and must therefore be assigned to the current level $\lambda>1$ *referer* partite set.

Recall that ***fBC*** already supports user preference algorithmically by allowing the user to root the level $\lambda>1$ biclique subgraph in his choice of a *next* page ($\mathbf{r_j}$) in line with his search interest. The ***3-pLM*** makes possible the identification of links (edges) that should be added in order to qualify pages of relevance for inclusion in the actual level $\lambda>1$ *request* partite set. In the experiment, the Web user did choose **http: $\mathbf{r_j}$ = manufacturers** to root the generation of the level $\lambda=2$ biclique subgraph. Column two of Table 10 lists a total of seven *request* pages referenced by the user's page of interest $\mathbf{r_j}$. The Web user should be given access to each of these *requests*. That is to say, a path from any of these *request* pages back to each and every level $\lambda=2$ must be constructed to genuinely promote user search customization. Experimental results captured in Table 11 are:

1. the 11 *requests* in the level $\lambda=2$ candidate *request* partite set (column 2),
2. α-adjacency results from Steps 33-38 excerpted from Table 7c (column 3),
3. the use of the tri-paginal link metric to determine inclusion (+) or exclusion (-) of a candidate request page (column 4), *and*
4. the α-distance $d_\alpha(\mathbf{r_j},\mathbf{rq_i})=1$, $\mathbf{r_j}$= **http:/music/machine/manufacturers/** and *request* pages $\mathbf{rq_i}$ in $\mathbf{CRQ^2} = \{\mathbf{rq_i} \mid \mathbf{1 \le i \le 11}\}$.

The detail characteristics of edges (links) between the user's preference page $\mathbf{r_j}$ and all *request* pages $\mathbf{rq_i}$ implicated by the *referer* partite set are evaluated by the ***3-pL*** metric for restructuring purposes that are implemented as described in item 3 above.

Table 11. Link analysis data chart of user selected page of interest r_j and pages assigned to level-2 candidate request partite set.

rq_i (1)	U R_j ={rq_i} (2)	r_j=manufacturers $r_j \rightarrow rq_i$, $r_j \leftarrow rq_i$ (3)	*3-pL* (4)	*#3-pL*	*distance* d_α= (5)
rq1	search.html	1, 1	+		1
rq2	schematics.html	1, 1	+		1
rq3	software.html	1, 1	+		1
rq4	prices	1, 1	+		1
rq5	mods.html	1, 0	(+)	1	1
rq6	images.html	1, 1	+		1
rq7	categories	1, 1	+		1
rq8	new.html	0, 1	(+)	6	1
rq9	images.html	1, 1	+		1
rq10	ecards	0, 0	-	0	($\neq$1)
rq11	addressbook	0, 0	-	0	($\neq$1)

Pages $\mathbf{rq_1}$ to $\mathbf{rq_7}$ are mandated members of the *request* partite set. Page $\mathbf{rq_1}$ has already been demonstrated to meet the α-adjacency criteria for inclusion since it was the only page in the intersection of the request sets, the $\mathbf{R_j}$s. How does the ***3-pL*** metric validate this finding? An outline detailing the use of the *3-pL* metric at biclique-lattice level λ>1 has been provided in Illustration 7.

First, set $\mathbf{rq_1}$**=U:/music/machine/search.html** as the nucleus of the ***3-pLM*** to determine if α-adjacencies exist between the nucleus and each page of the *referer* partite set at level-2. Figure 13 demonstrates use of the ***3-pLM*** in measuring the relevance of *request* page *search.html* role as a partite set member. Notice that no link deficiencies exist between *search.html* and pages in the *referer* partite set at level-2. Algorithm ***fBC*** and the ***3-pLM*** agree on the inclusion of *search.html* in the level-2 *request* partite set.

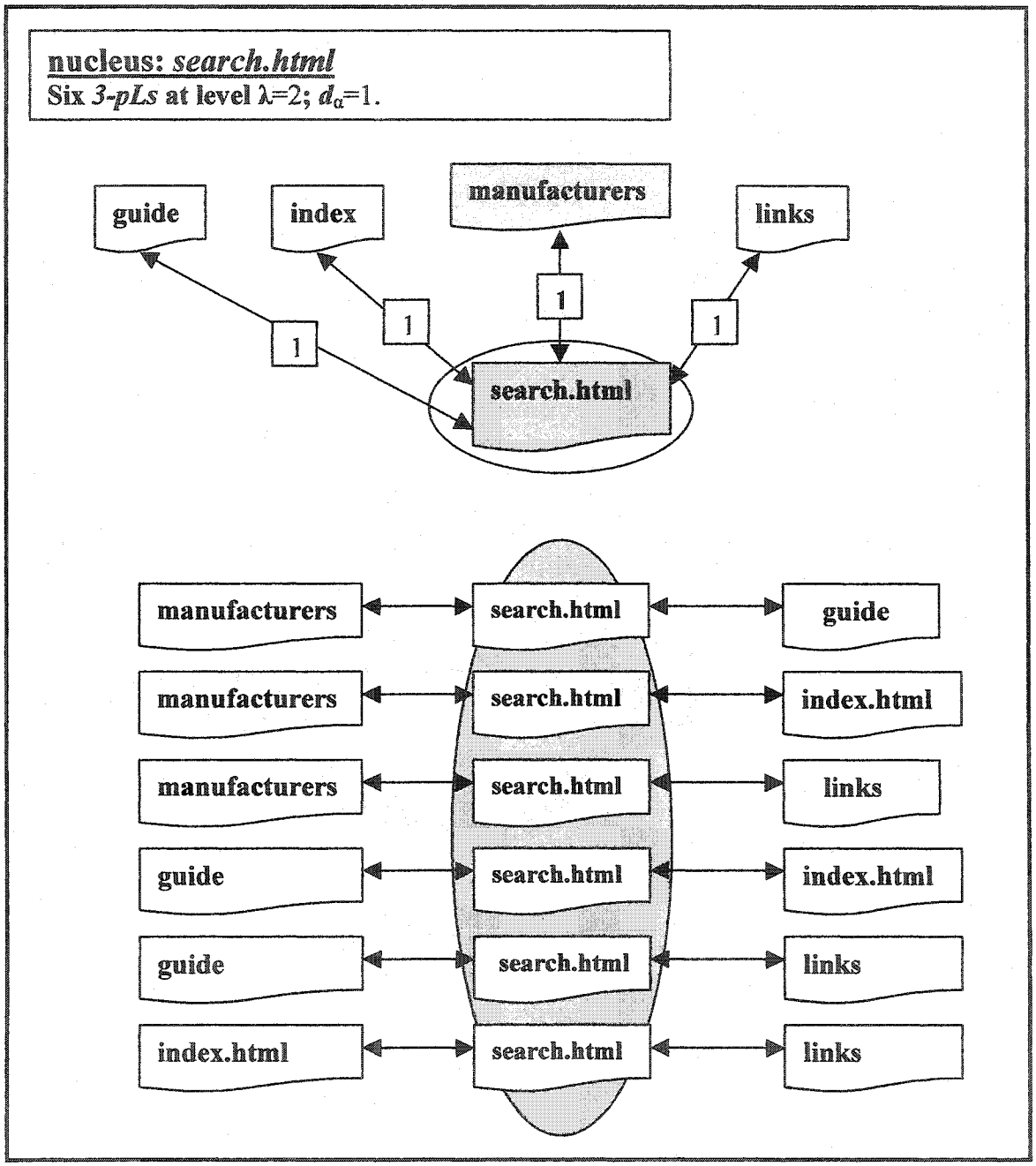

Figure 13. Link analysis using 3-pL metric indicates α-adjacency exists between *request* page 'search.html' and each level λ=2 *referer* partite set page.

Figure 14, Figure 15, and **Figure 16** demonstrate the application of the *3-pL* metric to *request* pages *mods.html*, *new.html* and *ecards*. These figures graphically highlight the *3-pLM*'s effectiveness in gauging the fitness of potential *request* page. The *3-pLM* also delivers exact and accurate data needed to qualify a relevant page for inclusion in the *request* partite set via restructuring by identifying and qualifying link deficiencies.

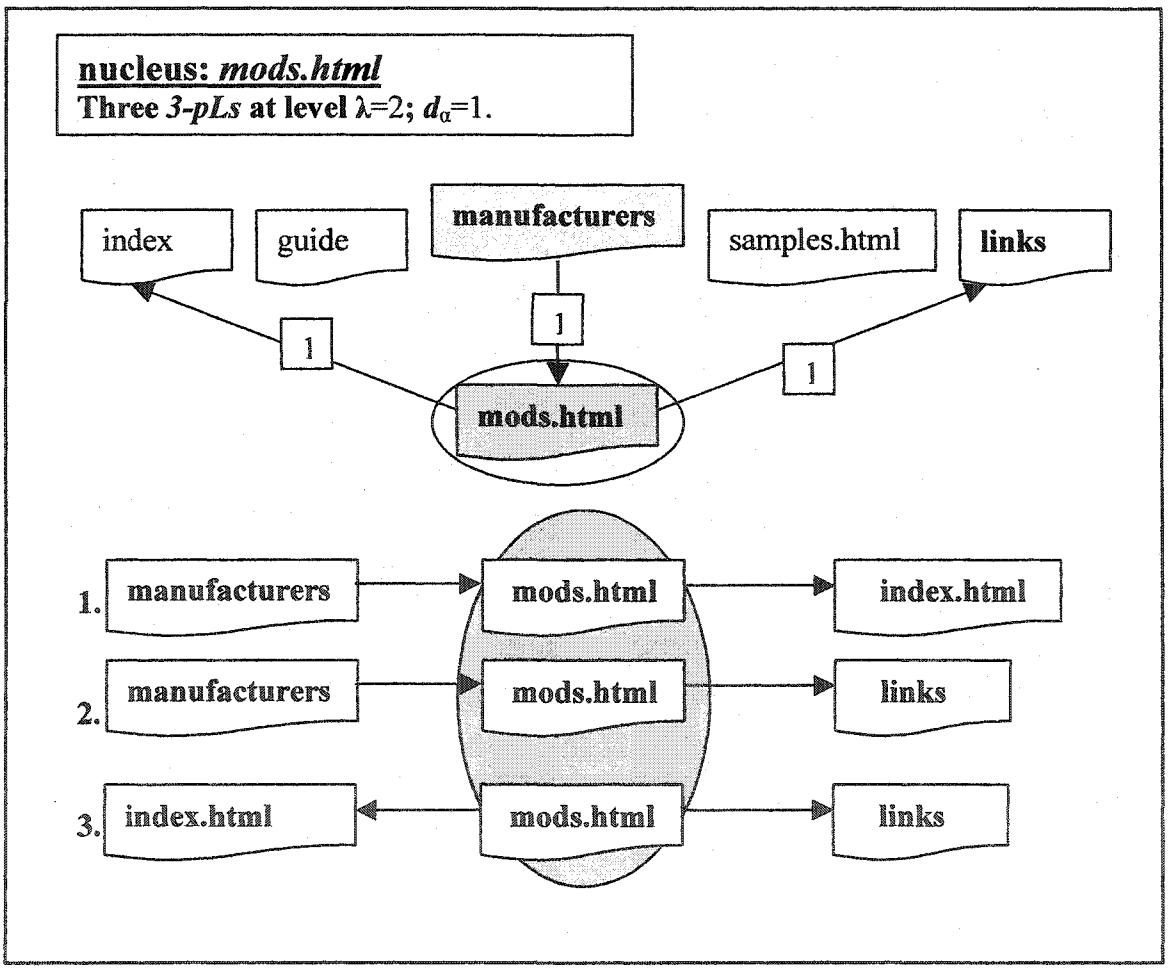

Figure 14. Tri-paginal link assessment of request page 'mods.html.'

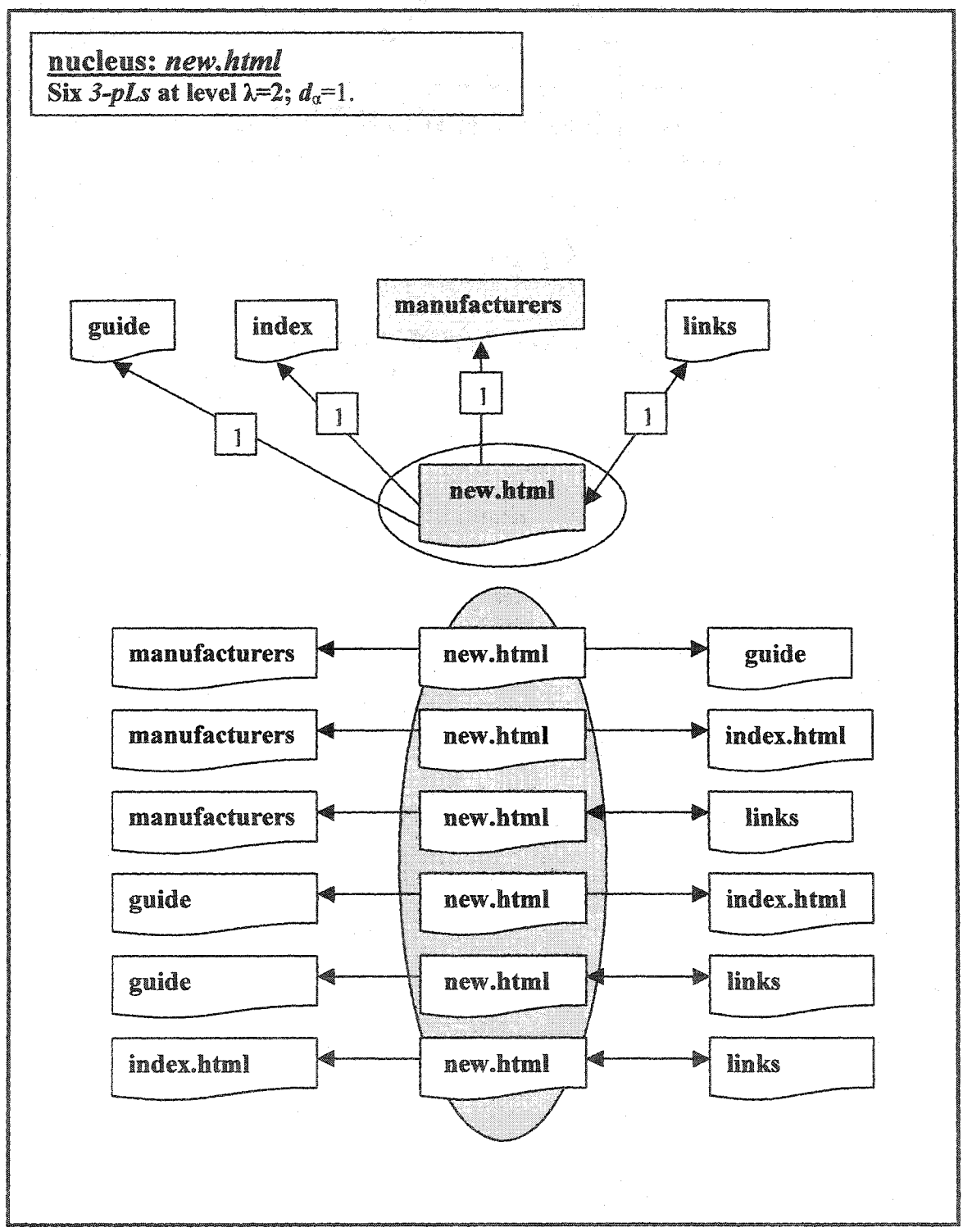

Figure 15. Tri-paginal link assessment of request page 'new.html.'

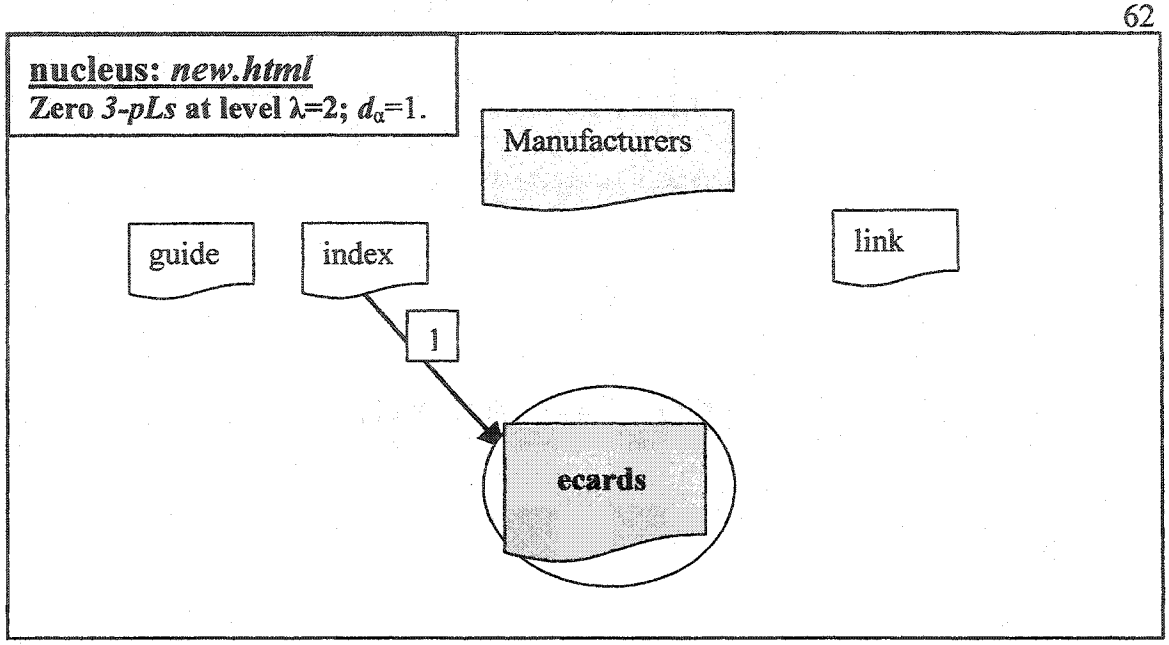

Figure 16. The *3-pLM* assessment of page *ecards* shows no thematic connection or relevance to the user selected page *manufacturers*. *Ecards* is identified for exclusion from the current candidate *request* partite set.

Figure 16 demonstrates Web page *ecards'* lack of fitness in the role of a level λ=2 *request* page in that no link exists between page *ecards* and the user's page of choice *manufacturers*. No ***3-pL*** with *ecards* serving as the nucleus can be instantiated. This means that page *ecards* is spatially and thematically distant from the user's search interest or preference as characterized by his level λ=2 choice of page $\mathbf{r}_1$. Hence, *ecards* is disqualified as a level-2 *request* partite set member.

If α-adjacency does not exist between a *request* page *rq* and the user's page choice $\mathbf{r}_j$ at level λ>1, then the ***3-pLM*** is used to first establish whether there is a distance of $\mathbf{d}_\alpha$=1 between the user's page choice $\mathbf{r}_j$ and the nucleus *request* page *rq*. This is to ensure thematic and spatial closeness between the *request* page and the user's indicated search interest. Since the *referer* partite set of pages have already been determined to be

relevant to the user's search objective at level λ=1, at least one *3-pL* structure that includes the user's page of interest $\mathbf{r}_1$ and the nucleus *rq* is required to qualify page *rq* for inclusion in the actual *request* partite set of the virtual biclique search neighborhood generated by algorithm *fBC*. Once the forgoing is established, then and only then can links be added to achieve α-adjacency between the *fit request* page and the *referer* partite page set at level λ > 1. A comparison of Figure 15 and Figure 17 graphically depicts how the *3-PLM* is used to identify page *new.html* as relevant and qualify it for inclusion in the *request* partite set at level λ=2. The link deficiencies observable in Figure 15 are overcome using restructuring activities (adding links) to produce α-adjacencies between the nucleus and each page in the current *referer* partite set. Figure 17 shows the results of restructuring.

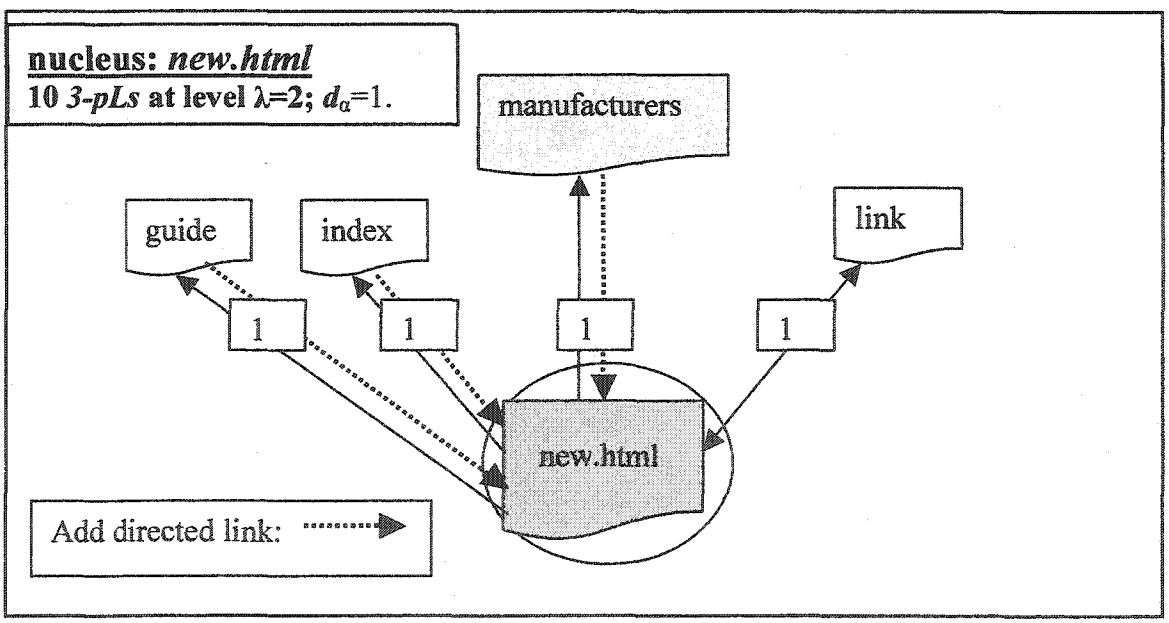

Figure 17. Restructuring activity of adding links is used to qualify page *new.html* for inclusion in the level 2 *request* partite set.

The level λ=2 biclique-lattice search neighborhood (pictured in Figure 18) produced by the ***fBC*** algorithm after incorporating the use of the ***3-pLM*** yielded the essential adjacency data needed to add links that expanded the level λ=2 *request* partite set by nine additional pages of value. The ***3-pL***s of *request* partite set pages with link deficiencies but having spatial qualities that strongly suggest thematic consistency (hub-authority dynamics) with the *referer* partite set were made accessible to the Web user via restructuring.

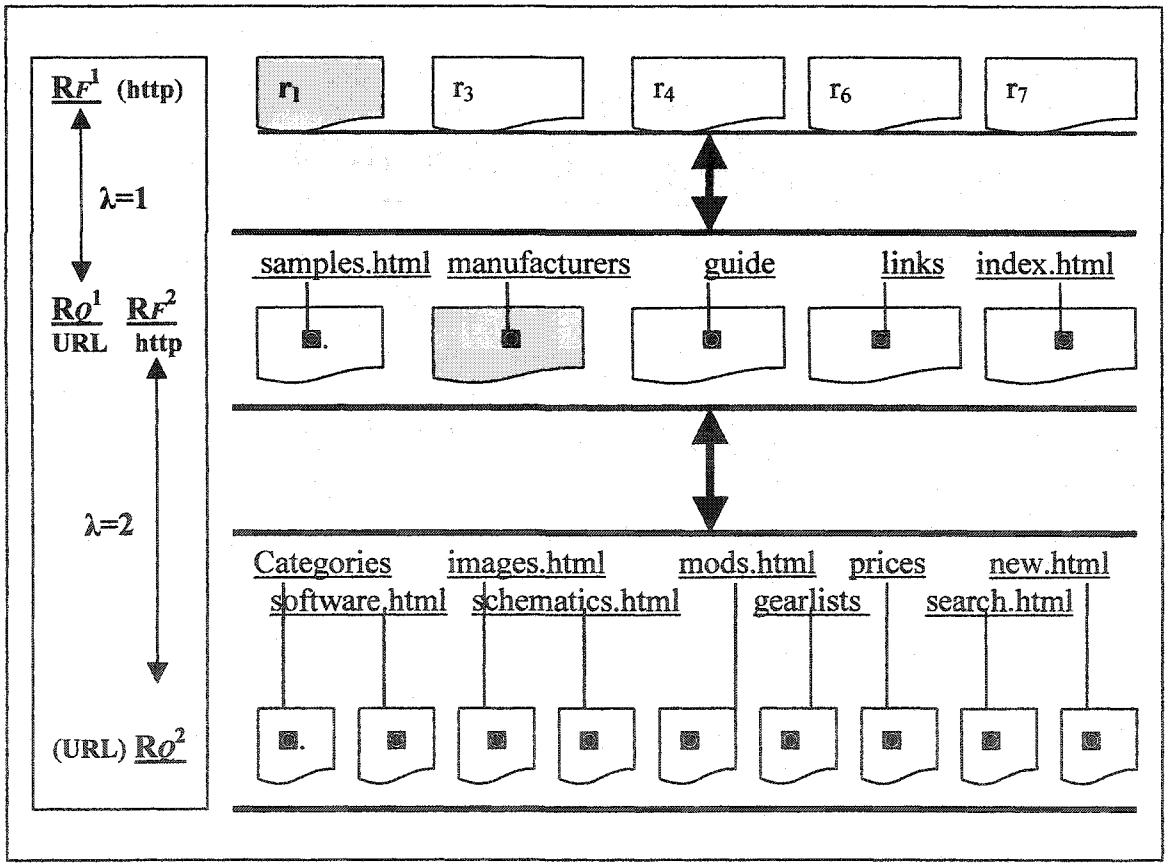

Figure 18. The level 2 biclique-lattice generated by *fBC* algorithm after the incorporation of novel *3-pL meter*.

set were made *fit* for inclusion in the level λ=2 *request* partite set. The ***3-pLM*** identifies relevant links to make each pages of value accessible to the Web user. The ***3-pLM*** identifies each relevant *request* page and makes it α-adjacent (add link restructuring) to each and every page in the *referer* partite set.

The ***3-pL*** meter facilitates the restructuring process of adding/deleting links in a structurally responsible way. The link is not added just because the user indicates his desire to access just any page, but it is added if it will reinforce the spatial and thematic structure of the strongly connected biclique-lattice search neighborhood which is dynamically created around the user's initial choice of a page which defines his search agenda.

3.4.4 Experimental Results

The methods and associated Web page metrics developed in this dissertation research to actualize Web user search customization yielded the following results when applied to the experimental log data set ***D*** containing 46,735 user access entries from Music Machines' archive (Web server access log data). Out of 1,140 distinct pages found in the data set, fourteen (14) meet the distance measure criteria of 'one' unit length from the Web user's initial page choice $r_{1=}$ http://www.hyperreal.com/music/machines/ or 'one' unit from the pages pointed to by the initial page choice. The methods identified the 14 pages listed in Table 12 as thematically relevant to the Web user's search objective. Four of the pages are ranked as authority pages which means that they are thematically similar to the user's initial page choice, and 9 pages of more specific topic content which support in a more topically

Table 12. Fourteen Web pages identified by *fBC* algorithm to be pages of relevance to the Web user's search rooted in r_1.

	d=1 from user's initial choice page r_1	Level λ
1	**U:/music/machine/samples.html**	1
2	**U:/music/machine/manufacturers/**	1
3	**U:/music/machine/guide/**	1
4	**U:/music/machine/links/**	1
5	**U:/music/machine/index.html**	1
6	**U:/music/machine/categories/**	2
7	**U:/music/machine/gearlists/**	2
8	**U:/music/machine/images.html**	2
9	**U:/music/machine/mods.html**	2
10	**U:/music/machine/new.html**	2
11	**U:/music/machine/prices/**	2
12	**U:/music/machine/schematics.html**	2
13	**U:/music/machine/search.html**	2
14	**U:/music/machine/software.html**	2

specific way the contextual framework of the user's search agenda since these 9 pages are directly linked to the authority pages which included the user's first page choice. The hierarchal arrangement of these 14 pages into a strongly connected virtual bipartite clique neighborhood generated by the methods used in this research is presented in Figure 18.

CHAPTER 4

SUMMARY/CONCLUSION

Web usage and Web structure mining were used in this work to design the ***fBC*** algorithm to create a closely connected virtual search neighborhood (at a host WWWebsite) organized around user preference. The restructuring component of the approach (which incorporates the novel tri-paginal link meter) is provided to ensure that the user's right to access pages of search relevance is guaranteed. Two maximal biclique search neighborhoods were initially extracted from the log data set ***D*** (Section 3.1) using page relevance ratings based on page adjacency, hit frequency, and the *Minimal Interest Level* metric.

4.1 Performance of *fBC* Algorithm and the Web Page Relevance Metrics

The ***fBC*** algorithm identified and organized 14 pages of value to the Web user's search objective. These pages are linked in an authority-hub relationship and are assigned correspondingly to the *referer* (authoritative pages) or *request* (hub pages) partite sets defining the biclique infrastructure of the search neighborhood.

To demonstrate that the 14 pages generated by the experiment performed in this work are indeed thematically and spatially organized around the user's preference, a comparison of results achieved by the highly respected PageRank algorithm [11] (which

is used by the Google search engine [17]) is used here to rank and order the 14 pages generated by this work's *fBC* algorithm (See Table 12). PageRank counts the number of back links (a subset of the 5,449 back links to the user's initial page choice) to each page in a page set (14 pages generated by *fBC*) to determine its importance or rank in a topically cohesive page set. Table 13 and Table 14 contain the hierarchal ordering achieved by *fBC* and the ordering done by *PageRank* using the data set *D* described in Section 3.1.

Table 13. Ranking related pages using algorithms *fBC* and PageRank.

	d=1 ***from user's initial page choice*** r_1=*http://www.hyperreal.com/music/machines/*	**Level** λ	***fBC*** **= 1 ... = 2**	**PageRank** ω
	URL:			
1	**U:/music/machine/samples.html**	1	1089	2113
2	**U:/music/machine/manufacturers/**	1	838	2586
3	**U:/music/machine/guide/**	1	694	1125
4	**U:/music/machine/links/**	1	320	889
5	**U:/music/machine/index.html**	1	77	526
6	**U:/music/machine/categories/**	2	0 (276)	547
7	**U:/music/machine/gearlists/**	2	0 (157)	206
8	**U:/music/machine/images.html**	2	448 (157)	718
9	**U:/music/machine/mods.html**	2	0 (41)	61
10	**U:/music/machine/new.html**	2	257 (*)	352
11	**U:/music/machine/prices/**	2	0 (69)	114
12	**U:/music/machine/schematics.html**	2	0 (198)	274
13	**U:/music/machine/search.html**	2	276 (43)	1188
14	**U:/music/machine/software.html**	2	0 (119)	157

PageRank rates page *search.html* as the third most authoritative or important page in the set of 14 pages, while methods and metrics defining the *fBC* algorithm eliminates this page from the level λ =1 biclique *referer* partite set which contains the most authoritative page, the user's initial page of choice, in the biclique neighborhood. This exclusion by

fBC means that the *search.html* page is of no thematic or spatial consequence to the user's initial page choice.

Table 14. PageRank's order of importance ratings compared to *fBC* hierarchal ordering of topically related Web pages.

	d=1 *from user's initial choice page r_1*	Level λ	*fBC* = 1 ... = 2	PageRank ω
2	**U:/music/machine/manufacturers/**	1	838	2586
1	**U:/music/machine/samples.html**	1	1089	2113
13	**U:/music/machine/search.html**	2	276 (43)	1188
3	**U:/music/machine/guide/**	1	694	1125
4	**U:/music/machine/links/**	1	320	889
8	**U:/music/machine/images.html**	2	448 (157)	718
6	**U:/music/machine/categories/**	2	0 (276)	547
5	**U:/music/machine/index.html**	1	77	526
10	**U:/music/machine/new.html**	2	257 (*)	352
12	**U:/music/machine/schematics.html**	2	0 (198)	274
7	**U:/music/machine/gearlists/**	2	0 (157)	206
14	**U:/music/machine/software.html**	2	0 (119)	157
11	**U:/music/machine/prices/**	2	0 (69)	114
9	**U:/music/machine/mods.html**	2	0 (41)	61

Table 15 gives us a closer look at the thematic and spatial attributes of the page that PageRank rated so high. Noticeably, the table shows no back-links from ***search.html*** to the most important authoritative page of all, the user's initial page of choice, **U:/music/machines/.** The criterion of α-adjacency is not met, hence *fBC* excludes it from participation at level λ = 1. A further look at *search.html* thematic ties to the user's initial page of choice shows that pages *Analogue-Heaven* and *email.html* are excluded from both levels of the biclique neighborhood, while the pages *guide, index.html, links and manufacturer* (that *search.html* points to) are included as pages of thematic and

spatial relevance to the user's initial page of choice. Hence, *search.html* presents itself at this point as a navigational convenience to pages of real consequence, all of which are included by ***fBC*** as pages of relevance to the user's initial page choice.

Table 15. The set of *requests* pages to which page *search.html* is a *referer*.

#	***URL:***	ω	*d*
1	U:/music/machines/Analogue-Heaven/	56	1
2	U:/music/machines/adaptive/browsing.html	7	2
3	U:/music/machines/adaptive/found.html	3	2
4	U:/music/machines/adaptive/notfound.html	22	2
5	U:/music/machines/email.html	1	1
6	U:/music/machines/guide/	31	1
7	U:/music/machines/index.html	14	1
8	U:/music/machines/links/	40	1
9	U:/music/machines/manufacturer/	72	1
10	U:/music/machines/search.html	10	1

The expertise (log access data) of the collective user with similar search agendas (traversing Web user's page of choice $\mathbf{r_l}$ which defines his search agenda) is used in a collaborative manner to assess current page value. Over 28% of the 46,735 entries in the data set ***D*** are *requests* for pages that the ***fBC*** algorithm selected as partite set members in the biclique search neighborhood it generates. Table 16 shows that 40% of the 11,404 log entries referring to pages in the initial candidate *request* partite set are pages referenced via the Web user's initial page of choice.

The ***Minimal Interest Level*** metric was applied to lessen the chance of eliminating relevant request pages whose low hit frequencies (e.g., *index.html*) indicate

otherwise, and exclude pages showing no immediate potential value to the Web user although a high hit frequency (e.g., *ecards*) may indicate relevance. The ***MinIL*** metric eliminated Web pages *ecards*, *email.html*, and *Analogue-Heaven* from participation in the level one *request* partite set.

Table 16. Hit frequencies of *request* pages with in-links from r_1 compared with their referral counts.

Rq#	Request URL U:/music/machine...	#Referrals by r_1 (ω)	#Referrals by ALL (ω)
rq1	/samples.html	1089	2113
rq2	/manufacturers/	838	2586
rq3	/guide/	694	1125
rq4	/images.html	448	718
rq5	/links/	320	889
rq6	/search.html	276	1188
rq7	/adaptive/	183	767
rq8	/ecards/	134	189
rq9	/index.html	77	526
rq10	/email.html	9	80
rq11	/Analogue-Heaven	2	1223
***	TOTAL	4070	11404

The spatial and thematic organization of pages enforced by the virtual biclique infrastructure of search neighborhood has implications for structuring and/or restructuring Web host sites. The hierarchal management of Web pages facilitated by the virtual biclique search neighborhood provides the Website designer a dynamic knowledge base for improving Web user satisfaction relative to quality of page content, page location, and site navigation. (At this point, it is suggested that Table 2, Table 12, and Table 16 used).

during the following discussion). Notice that both *ecards* and *Anlogue-Heaven* were eliminated as pages of value at both levels of the biclique search neighborhood even though they were selected initially as potential pages of relevance because they met the spatial criteria of $d_\alpha = 1$ relative to the user's initial page choice. Interestingly, a close look at *ecard* shows that it defines the picture images of the icons displayed on the user's initital page of choice. It is obvious that this page is not thematically tied to the information sought by the Web user and therefore should not have been included as a page of topical relevance to the search. Likewise, the *Analogue-Heaven* page was eliminated because no thematic connection to the user's search objective was found. The access log data set used in this research shows that 53.4% of the 1,223 hits on *Analogue-Heaven* are re-directed accesses to the Web user's initial page choice $\mathbf{r_1}$. The methods and metrics developed in this research detected neither α-distances or β-distances between the *Analogue-Heaven* page set and any of the pages of relevance to the Web user's search objective. This goes to verify that the *Analogue-Heaven* page set is disjoint from the set containing all of the other pages since visitors apparently use the *Analogue-Haven* page as a navigational convenience to access the user's initial page of choice and other pages not in the *Analogue-Heaven* page set. A closer look at *Analogue-Heaven* shows that it is an archive of self-contained pages dealing with analogue music machine devices. Only 570 (1.2%) hits on this page set out of the 46,735 total hits in the data set further substantiate its thematic distance from the user's search interest and that it should not have been included as a page of relevance to the Web user's search agenda.

4.2 Recommendation: Format of Web Server's Access Data Log

Attention is called to the ***combined log format*** with its *referer* and *request* parameters as an invaluable aid in the assessment of spatial and contextual/topical relatedness of a Website's resources (pages/objects). A salient feature of the ***fBC*** algorithm is that it maintains the structural integrity of the dynamically generated virtual biclique search neighborhood due to the linkage data it derives from the log access entries in C_bLF.

It is recommended that Website administrators give serious consideration to a change from the most popular and commonly used Common Log Format to the Combined Log Format for the collection of Web server access data logs. Web structure mining performed on log data set ***D*** has been very effective in demonstrating the value of linkage data (inherent in the C_bLF access log) to the discovery of thematic and spatial ties between Web pages, and by extension the discovery of a Website's structural strengths and weaknesses of a spatial and contextual type. The Web designers and administrators can readily use this knowledge to make decisions relative to page content, addition/deletion of pages, or relocation of pages for the purpose of enhancing the Website's appeal by improving its resource offerings and navigational quality.

4.3 Direction for Future Research

The approach in this dissertation has demonstrated success in identifying and organizing pages that are thematically and spatially related to the Web user's search objective and organizing them into a strongly connected neighborhood. However,

interest in the benefits of using the temporal data (timestamp is available in the access log) in combination with usage and linkage data to establish page relevance would seem to be a direction for further research in Web user customization. This consideration is prompted by the following example. Let p_1 and p_2 be two α-adjacent pages with similar high hit frequencies and thus selected as pages of value. Suppose that timestamp data indicates that on average visitors spend 10 seconds at p_1 while visitors spend on average 10 minutes accessing p_2. The temporal parameter could possibly allow for determining whether p_1 and p_2 should be attributed equal status as pages of relevance or might p_1 be just a navigational convenience linked directly to p_2.

REFERENCES

[1] B. Yuwono and D. Lee, "Search and Ranking Algorithms for Locating Resources on the World Wide Web," *Proceedings of the 12th International Conference on Data Engineering*, March 1966.

[2] in *New York Times, electronic version*, November 11, 2004.

[3] M. H. Dunham, *Data Mining - Introductory and Advanced Topics*, 1 ed: Prentice Hall, 2003.

[4] R. Kumar, P. Raghavan, P. Raghavan, D. Sivakumar, A. S. Tomkins, and E. Upfal, "The Web as a Graph," in *Proceedings of the 19th Annual ACM SIGMOD SIGACT-SIGART Symposium on Principals of Database Systems*: ACM, May 2000.

[5] D. B. West, *Introduction to Graph Theory*, 2nd ed: Prentice Hall, 2001.

[6] R. Kumar, P. Raghavan, S. Rajagopalan, and A. S. Tomkins, "Trawling the Web for Emerging Cyber-Communities," *Proceedings of the Eight World Wide Web Conference (WWW8)*, April 1999.

[7] M. Toyoda and M. Kitsuregawa, "Creating a Web Community Chart for Navigating Related Communities," *ACM*, 2001.

[8] R. Kosala and H. Blockeel, "Web Mining Research: A Survey," *SIGKDD Explorations*, vol. 2, pp. 1-15, July 2000.

[9] S. K. Rangarajan, V. V. Phoha, K. S. Balagani, R. Selmic, R., and S. S. Iyengar, "Adaptive Neural Network Clustering of Web Users," *IEEE Computer Society*, pp. 34-40, 2004.

[10] M. Eirinaki and M. Vaziriannis, "Web Mining for Web Personalization," *ACM Transactions on Internet Technology*, vol. 3, February 2003.

[11] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank Citation Ranking: Bring Order to the Web," *http://google.stanford.edu/back-rub/pagerranksub.ps*, 1998.

[12] S. Chakrabarti, B. Dom, P. Raghavan, S. Rajagopalan, D. Gibson, and J. Kleinberg, "Automatic Resource Compilation by Analyzing Hyperlink Structure and Associated Text," *Proceedings of the 7th International World Wide Web Conference*, 1998.

[13] D. Gibson, J. Kleinberg, and P. Raghavan, "Inferring Web Communities from Link Topology," *Proceedings of HyperText98*, 1998.

[14] R. Kumar, P. Raghavan, S. Rajagopalan, and A. S. Tomkins, "Extracting Large-Scale Knowledge Bases from the Web," *Proceedings of the 25th VLDB Conference*, 1999.

[15] R. Cooley, "The Use of Web Structure and Content to Identify Subjectively Interesting Web Usage Patterns," *ACM Transactions on Internet Technology*, vol. 3, pp. 93 - 116, May 2003.

[16] M. N. Garofalakis, R. Rastogi, S. Seshadri, and K. Shim, "Data Mining and The Web: Past, Present and Future," *Proc. 2nd Int'l Workshop Web Information and Data Management, ACM Press*, pp. 43-47, February 1999.

[17] Google.*www.google.com/press/pressrel/3billion.html*, December 2001.

[18] M. Toyoda and M. Kitsuregawa, "Tools for Organization: Creating a Web Community Chart for Navigating Related Communities," *Proceedings of the 12th ACM Conference on Hypertext and Hypermedia*, Sept 2001.

[19] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. S. Tomkins, and J. Wiener, "Graph Structure in the Web," *http://www9.org/w9cdrom/160/160.html*, 2000.

[20] M.-S. Chen and J. S. Park, "Efficient Data Mining for Path Traversal Patterns," *IEEE Transactions on Knowledge and Data Engineering*, vol. 10, pp. 209-231, 1998.

[21] J. Wang, Z. Chen, L. Tae, and W.-Y. Ma, Wenyin, Liu, "Ranking User's Relevance to a Topic through Link Analysis on Web Logs," *Proceedings of the 14th International Workshop on Web Information and Data Management*, pp. 49-50, 2002.

[22] J. M. Kleinberg, "Authoritative Sources in a Hyperlinked Environment," *Journal of the ACM*, pp. 604-632, September 1999.

[23] B. C. Miles and V. V. Phoha, "The Bipartite Clique - A Toplogical Paradigm for WWWeb User Search Customization," *Accepted (1/14/05) for publication in Proceedings of 43rd ACM Southwest Conference*, March 2005.

[24] S. Chakrabarti, B. Dom, R. S. Kumar, P. Raghavan, S. Rajagopalan, A. S. Tomkins, D. Gibson, and J. M. Kleinberg, "Mining the Webs Link Structure," *Computer*, vol. 32, pp. 60-67, August 1999.

[25] Y. Xie and V. V. Phoha, "Web User Clustering from Access Log Using Belief Function," *ACM*, 2001.

[26] J. Srivastava, R. Cooley, M. Deshpande, and P. N. Tan, "Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data," *SIGKDD Explorations*, pp. 12-13, January 2000.

[27] S. Dill, R. Kumar, K. S. Mcclurley, S. Rajagopalan, D. Sivakumar, and A. S. Tomkins, "Self-Similarity in the Web," *ACM Transactions on Internet Technology*, vol. 2, pp. 205-223, August 2002.

[28] M. Perkowitz and O. Etzioni, "Adaptive Websites: Automatically Synthesizing Web Pages," *American Association for Artificial Intelligence*, 1998.

[29] Y. Xiao and M. H. Dunham, "Efficient Mining of Traversal Patterns," *Data and Knowledge Engineering*, vol. 39, pp. 191-214, November 2001.

[30] D. Dhyani, W. K. Ng, and S. S. Bhowmick, "A Survey of Web Metric," *ACM Computing Surveys*, vol. 34, pp. 469-503, December 2002.

[31] J. Callender, *Perl for Website Management*, 1 ed: O'Reilly and Associates, Inc, 2001.