HIS'02: Proc. of the 2002 Hybrid Intelligent System, IOS Press (Dec. 2002) 891-900.

Simulating an Information Ecosystem within the WWW

Reginald L. Walker
Tapicu, Inc., P.O. Box 88492
Los Angeles, California 90009
rwalker@tapicu.com

Abstract. The design focus of the Tocorime Apicu integrated search engine builds upon new approaches and techniques associated with evolutionary computation to improve the precision and recall mechanisms of existing information retrieval systems within popular search engines. The interactions of the four major components of engines are facilitated through the use of a hierarchical communication topology which partitions the nodes of a distributed computing system into subclusters. The hierarchical communication topology is based on an information ecosystem modeled upon and incorporating the social structure of honeybees—this providing mechanisms for the efficient sharing of information.

1 Introduction

The information sharing/communication model associated with Tocorime Apicu [1] research effort [18, 17, 16] incorporates aspects of the unique behavior exhibited by inhabitants of a honeybee colony [9, 11] in relation to their external environment, including their relation to other honeybee colonies. These aspects are employed to adequately search and index portions of the Web for valuable information by viewing the WWW as an information ecosystem. Ultimately, resulting in a comprehensive event manager (EM) model [2], the honeybee model [14] removes communication limitations inherent in current methodologies by providing the basis for information sharing mechanisms. This model can be extended by treating each subcluster—based on the nodes associated with each manager, M_i, in the EM model—as a set of queens and drones.

Various techniques are employed to continuously disperse foragers within the honeybee information sharing model to serve the needs of an existing colony in their search for the location of ever-changing food sources (time-dependent information) which are prone to change drastically over a relatively short period of time in a manner similar to requesting information from a remote site within the Internet as shown in Figure 1. Web page retrieval is accomplished within the Tocorime Apicu engine by the HTML Resource Discovery (HRD) system [16] and Web page parsing by the Information Sharing Indexing (ISI) system [18, 17]. The foragers mark food sources as well as the path to the food source in order to formulate customized routes [16, 13, 10]. This methodology has the ability to discover new ISPs as well as new sub-host providing services to new and existing Web clients which result in faster discovery of new and updated Web pages.

[1] The word $Tocorime$, meaning "spirit"[7], comes from an ancient Amazon Indian language. $Apicu$ comes from the Latin $apis\ cultura$, meaning "honeybee culture" or "the study of honeybees." The phrase $Tocorime\ Apicu$ is used as "in the spirit of bee culture."

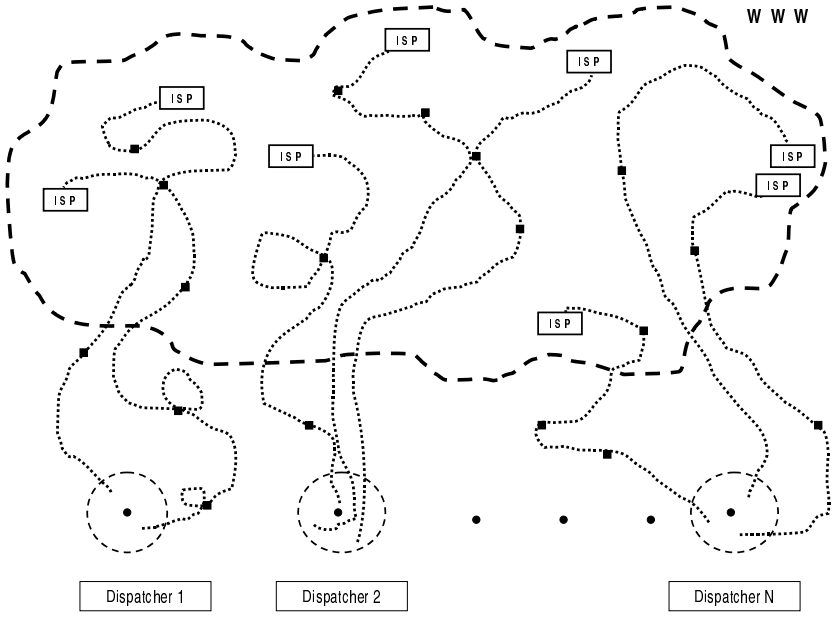

Figure 1: The WWW as an information ecosystem as viewed by the Tocorime Apicu HRD system Web dispatchers.

2 The Problem

Most of the current engines [12] provide its users with varied results—this depending on the type of producer—this in turn leading to the coupling of link analysis, human-based categories, and automated spidering technology. The result is overlapping pages/information. Several of these engines are producers as well as consumers of pre-compiled sets of Web pages. The unique qualities of crawler-based systems coupled with the results of human-editors provides an assortment of information sources, this further aggravating the task of indexing for the consumer engines which attempt to eliminate page duplication in the query results. Inevitably, this process should lead to diversity which may be apparent in the query results associated with various search strings.

The use of link analysis—which is based on the Web pages having been clicked/accessed by various past users of particular engines—allows various servers to handle different categories reflective of regional user interest. Such an approach is the underlying focus of Google [8] where the shortcomings of link analysis [12] are based on Web page connectivity which has been predicted as belonging to one of these categories:

1. Core pages—30 percent of the Web pages are considered interconnected and easy to find by their Internet connection links.

2. Originating pages—24 percent of the Web pages cannot be accessed from the core pages but are linked to them.

3. Destination pages—24 percent of the Web pages can be accessed from the core pages but

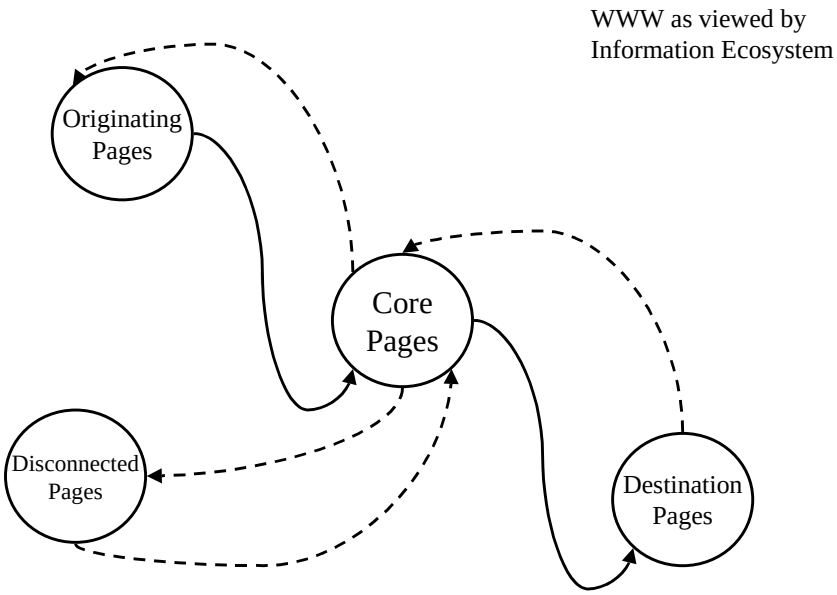

Figure 2: Web page connectivity as viewed by the Tocorime Apicu information ecosystem. Solid lines show Web page connectivity as viewed by Google.

are not linked to them.

4. Disconnected pages—22 percent of the Web pages are disconnected from the core pages.

3 The Simulate Honeybee Ecosystem

3.1 The Information Ecosystem

The creation of an information ecosystem, which emulates selected aspects of distributed honeybee colonies, incorporates three distinct hierarchical levels of biological organization [4, 5, 19] using the methodologies of evolutionary computation (EC) [17] and EM: the population, the organismic, and the genetic levels. The EM model provides the means by which the various interactions within and between levels of the ecosystem can take place.

The restrictions placed on the ecological system were: 1) the process of evolutionary search must agree with biological facts, even if non-biological search techniques are more effective, and 2) the information ecosystem must be as simple or primitive as possible. These restrictions correspond to the evolutionary processes incorporated in biological species, enabling them to solve problems typified by chaos, chance, temporality, and nonlinear interactivity [5, 3]. Figure 2 presents Web page connectivity as viewed by the Tocorime Apicu information ecosystem.

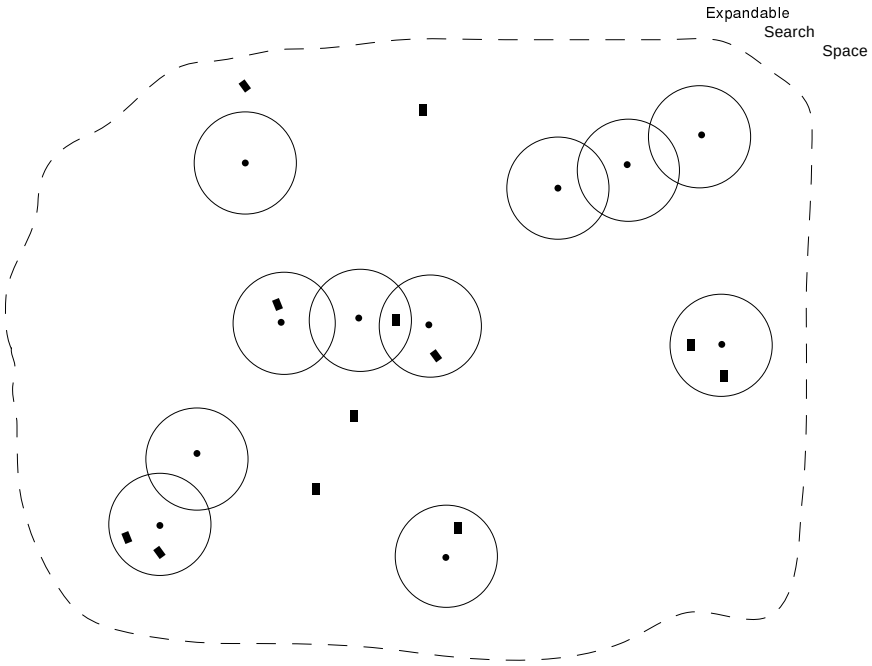

Figure 3: A snapshot of the expandable search space for overlapping NNCs as viewed by the ISI system.

3.2 The Hierarchical Levels of Honeybees

The social hierarchy of honeybees [11, 9] results from environmental influences—ecological and physiological, genetic potentiality, and social conditions (regulations) of the colony. Social insects are characterized by 1) cooperation among adults in brood care and nest construction, 2) overlapping of at least two generations, and 3) reproductive division of labor—queen specializes in egg laying and workers in brood care. The evolutionary search strategy of this system initiates unique characteristics of the honeybee model—its spatial organization along with species specifity, this in turn influencing the pattern of gene flow, and therefore the search strategy.

The methodology of EC provides mechanisms to facilitate the requirements needed to model the organic evolution of purposive behavior [6] which incorporates various aspects of honeybees. Organic evolution requires: 1) a system of stored information, 2) a system of transferring information from one generation to the next, 3) a system of mutation of stored information, and 4) a system of translations of the genetic information into a form against which value judgments are made. Purpose behavior has two aspects which are: 1) the ability to specify an intrinsic purpose and 2) elaborating a pattern of behavior that is directed towards achieving a specified purpose. The evolution of purposive behavior is achieved by requiring 1) stochastic fluctuations to break the impasse of indeterminacy and 2) the actions associated with the selection operator. The outcome of this process is a deterministic sequence of events resulting in a stochastic process which sets a basically indeterminate system into one of many possible pathways.

Table 1: Cumulative access log summaries for Web probe dispatchers.

Item	Web probe dispatchers
Start date	15 Oct 2001
Stop date	28 Jan 2002
Duration in days	105 days
Duration in weeks	15 weeks
Simultaneous probe transmissions	128
Total requests	48193332.0
Avg requests per day	458984.1
HTML servers located	75367

4 Overview of the Tocorime Apicu Information Sharing (IS) System

4.1 Web Page Indexing Components

Restricted and free breeding were implemented by the formulation of an expandable search space (Figure 3) which forms adaptive subclusters. The search space shown in the figure depicts a snapshot of the location of drifter nodes with respect to seed nodes. The seed nodes are represented as small circles and the drifters as small rectangles. Each specie seed represents the basis of a subgroup if it exists. A subgroup is formed when one or more drifters are within a given radius of a seed forming nearest neighbors. Each seed uses the same radius. An extension of this approach better reflects the biological model, allowing the seed radius to increase/decrease based on the decay of its pheromones [13, 10]. Supersedure emulation occurs when two or more seeds are nearest neighbors forming overlapping nearest neighbor clusters (NNCs). When a single drifter is a nearest neighbor of two disjoint seeds, this node may be selected to share information twice based on the existence of one or more competing drifters.

4.2 Web Foraging Components

The Web probes, scouts, and foragers are responsible for retrieving external data. The raw external data files are reduced to raw HTML files in the first pass that contain no padding or HTML header information [15], such as date of last modification or date to expire. The resulting raw HTML file is equivalent to the information presented to the user by her/his chosen Web browser. The raw external file contains information that can be used by the advanced mechanisms, which, themselves are components of some popular browsers.

The second pass on the raw HTML file occurs during the tag parsing process; this is a component of the ISI system. The data to be used in these mechanisms is filtered during the first pass in order to assure that the appropriate wrappers exist. Wrappers used by the HRD foragers filter out documents that may not be in English or follow incorrect HTML format. Currently, the search engine is unable to perform any language translations. The HRD dispatchers communicate with the ISI dispatchers via their respective group managers. Likewise, the ISI dispatchers communicate with the browser reporting interface (BRI) dispatchers through their respective manager interfaces.

Table 2: Stochastic measures associated with Web probe dispatchers.

Week	f_{loc}	f_{succ}	P(M,i)	R(z)	I(M,i,z)
1	0.04548	0.01734	0.00004	115127.0	230253.9
2	0.03687	0.01312	0.00003	153503.4	460510.1
3	0.03962	0.01610	0.00004	115127.0	460507.8
4	0.04210	0.01705	0.00004	115127.0	460507.8
5	0.04194	0.01595	0.00004	115127.0	690761.7
6	0.04078	0.01361	0.00003	153503.4	1074523.6
7	0.04025	0.01309	0.00003	153503.4	1228027.0
8	0.04184	0.01595	0.00004	115127.0	1036142.6
9	0.05023	0.01819	0.00003	153503.4	1535033.7
10	0.03905	0.01463	0.00005	92101.1	1013112.1
11	0.04024	0.01502	0.00004	115127.0	1381523.4
12	0.04035	0.01683	0.00005	92101.1	1197314.4
13	0.04244	0.01702	0.00005	92101.1	1289415.5
14	0.04215	0.01687	0.00004	115127.0	1726904.3
15	0.03919	0.01599	0.00004	115127.0	1842031.2
Mean	—	—	—	120755.5	1041771.3
SD	—	—	—	21612.3	475118.8

5 Experimental Results

5.1 HRD Experimental Environment

The HRD system searched the Internet for those ISPs hosting Web services [17] for a total of fifteen weeks which included three U.S. holidays—Thanksgiving, Christmas, and New Years (see Table 1). The start date was 15 October 2001 and the terminating date was 28 January 2002 with one-week data collection periods that span from Monday to Monday.

The goal of this study was to test the run-time environment associated with dispatchers and determine the limitations in executing the HRD network probing software for extended periods of time. The HRD system was tested using HP Pavilions with seven 733MHz (20 Gigabytes of memory) and one 766 MHz (30 Gigabytes of memory) Intel Celeron processors, 128 MB SDRAM, and Intel Pro/100+ Server Adapter Ethernet cards, connected via two D-Link DSH-16 10/100 dual speed hubs with switches through a 144 Kbps router. The dispatcher tests were run using Red Hat Linux release 7.0 (Guiness).

5.2 Web Probe Dispatchers

Usage of stochastic measurements [16] provides insight into the efficiency of the system and thus reflects performance degradations. The localized fitness, f_{loc}, of each version of the probe dispatcher per week is presented in Table 2. The number of independent runs required to achieve an efficiency of 99%, $R(z)$, showed moderate variations. The standard deviation for $R(z)$ and $I(M, i, z)$ was 21612.3 and 475118.8, respectively. The values associated with f_{loc} are dependent on the number of probes per week, thus reflecting the collisions within the LAN and TTL associated with the released probes. The success fitness, f_{succ}, which is also based on the weekly sum of all four dispatchers, determines the effectiveness of locating ISPs that host HTML services.

Table 3: Web scout dispatcher results for customized routing.

ISP response results	Web scout dispatcher				Totals
	Node 0	Node 1	Node 2	Node 3	
Number of DNS name resolutions (dispatcher result code Y)	9262	6774	6821	5870	28727
Number of QoS requests (based on RS statistic plots)	9502	9180	9587	9326	37595
Periodic tasks (recurring QoS requests)	240	2406	2766	3456	8868
Number of customized routes (successful retrievals)	1422	1127	1003	1112	4664

The corresponding number of projected dispatchers, $I(M, i, z)$, is not computed as a value independent of the future collection period. The value projects the number of dispatchers required to correct shortcomings of the current dispatchers. The number of required weeks, $R(z)$, needed to achieve the targeted probability was 92.1 thousand weeks for collection periods 11, 13, and 14. A projected value of 115.1 thousand weeks is required for collection periods 2, 4, 5, 6, 9, 12, 15, and 16. The remaining collection periods required 153.5 thousand weeks. The number of dispatchers needed for collection period 2 was projected as 0.2 million. Collection periods 3, 4, and 5 will require 0.5 million dispatchers. Collection periods 7, 8, 9, 11, 12, 13, and 14 will require between 1.0 and 1.5 million dispatchers. Collection periods 10, 15, and 16 will require between 1.5 and 1.9 million dispatchers. These results show some consistency between the distinct collection periods. This suggests that increasing the LAN throughput to the Internet should reduce all projected collection periods and dispatcher requirements.

5.3 Web Scout Dispatchers

The RS statistic results in the Table 3 differ from the references used as the basis of this approach [16] since the larger RS statistics value implies a smaller degree of self-similarity. RS statistics measurements are periodically used to determine the degree of congestion along the path to a chosen ISP, as well as within the chosen ISP. The retrieval of raw data files removes the ISP from the lists of periodic tasks which are then retested. The feasibility tests are based on the release of scouts to each ISP in order to perform the self-similarity computations. A congestion threshold was used to determine feasibility results, indicating 4544 customized routes. All of the customized routes contained at least one scout that did not return within its allocated TTL. These time-outs reflect possible congestion in the LAN, LAN+WAN, or WAN.

The scout dispatcher results are shown in Table 3. The total number of DNS name resolutions 28727. The increase is due to the extended collection period and dispatcher improvements [16].

Table 4: ISP responses to Web foragers inquiries based on customized routing.

Week	ISP response results	Web forager dispatcher				Totals
		Node 0	Node 1	Node 2	Node 3	
Week 1-15	Number of customized routes (retrieval of raw data files)	9262	6774	6821	5870	28727
Week 16	Raw HTML pages	690	536	548	557	2331
	Access forbidden pages					
	—Firewall pages	21	27	44	31	123
	—Web mail pages	110	144	129	155	538
	—403 forbidden pages	4	3	5	4	16
	—404 not found	0	0	2	2	4
	Useful raw HTML pages	555	362	368	365	1650
Week 17	Raw HTML pages	732	591	455	555	2333
	Access forbidden pages					
	—Firewall pages	75	10	28	17	130
	—Web mail pages	214	144	216	173	747
	—403 forbidden pages	3	1	0	1	5
	—404 not found	1	1	0	1	3
	Useful raw HTML pages	439	435	211	363	1448
Totals	Raw HTML pages	1422	1127	1003	1112	4664
	Access forbidden pages					
	—Firewall pages	96	37	72	48	253
	—Web mail pages	324	288	345	328	1285
	—403 forbidden pages	7	4	5	5	21
	—404 not found	1	1	2	3	7
	Useful raw HTML pages	994	797	579	728	3098

5.4 Web Forager Dispatchers

Use of the customized routes produced raw data files which were not always useful. The raw HTML pages were the results of the file being validated by pre-parsing the raw contents, which are a component of the ISI system. The pre-processing may result in NULL files which were discarded, thus indicating that either the HTML format was incorrect, or that the designer used non-English tags. Other non-useful pages indicated a firewall or Web mail login (access forbidden) HTML page but were retained in these studies.

Table 4 presents the corresponding ratios: 15.4%, 16.6%, 14.7%, and 18.9% for the forager dispatchers, respectively. The cumulative dispatcher ratio for this version was 16.2%. The results reflect the use of filters that eliminated a host of raw HTML pages through implementation of the first pass of the two-pass parser [18] required by the Tocorime Apicu ISI system.

6 Related Work

A search algorithm termed “optimization with marriage in honeybees” (MBO) [1] emulated the mating behavior of bees. The simulated behavior in this approach relied on the in-flight aspect of the mating-flight of queens. The variable-speed flight of the queen was viewed as a set of transitions in a state space in which the queen mates with a different drone at each state

probabilistically.

By effectively sharing/disseminating information among individuals, swarm strategies [14] incorporate mechanisms to solve optimization problems. This approach incorporates species' seeds which form the nucleus of a collection of random individuals. Each individual swarm member communicates with its nearest neighbors (NN) in order to improve their performance and obtain the information required in searching for the optimum among/in smaller groups (also known as afterswarms in honeybee terminology).

Selection pressure on subpopulations is reduced when the "migration interval" for an application is random. This approach facilitates implementation of the methodology of the SBGA [19] algorithm. The continuous creation of subpopulations leads to various combinations of individuals with similar genetic makeups. This results in shifts in the gene space, which, in turn, facilitate the exploration of the disparate regions of the search space.

An ant colony optimization (ACO) approach [13, 10] for the resource constrained project scheduling problem (RCPSP) was implemented to simulate the pheromone trails associated with each ant. The use of the pheromone trail required the implementation of heuristics that determine the rate of pheromone decay (evaporation).

7 Conclusion

The creation of an artificial ecosystem within the Internet provides an adaptive search model which has the ability to evolve and keep pace with the Internet's evolution and further development. The exponential growth of related Web documents presents IR problems for existing and newly developed engines. The IR problems are magnified as the existing engines set new goals for incorporating documents that do not comprise the group of pages considered the core of Internet related documents. The communication patterns within this simulated system benefit from the adaptive communication and hierarchical habits of honeybees.

8 Acknowledgments

The author wishes to thank D. Stott Parker and Gary B. Fogel for their direction and suggestions. The author wishes to express his gratitude to the reviewers whose detailed and useful comments helped tremendously to improve the quality of this paper. This work was supported by Honeybee Technologies and Tapicu, Inc.

References

[1] H.A. Abbass. MBO: Marriage in Honey Bees Optimization A Haplometrosis Polygynous Swarming Approach. In *Proceedings of CEC 2001*, pages 207–214. IEEE, Piscataway, NJ, 2001.

[2] R. Bagrodia. Process Synchronization: Design and Performance Evaluation of Distributed Algorithms. *IEEE Transactions on Software Engineering*, 15(9):1053–1064, September 1989.

[3] P.J. Bentley, T.G.W. Gordon, J. Kim, and S. Kumar. New Trends in Evolutionary Computation. In *Proceedings of CEC 2001*, pages 162–169. IEEE, Piscataway, NJ, 2001.

[4] M. Conrad and H.H. Pattee. Evolution Experiments with an Artificial Ecosystems. *Journal of Theoretical Biology*, 28:393–409, 1970.

[5] D.B. Fogel. An Introduction to Simulated Optimization. *IEEE Transactions on Neural Networks*, 5(1):3–14, 1994.

[6] A.S. Fraser. The Evolution of Purposive Behavior. In H. von Foerster, J.D. White, L.J. Peterson, and J.K. Russell, editors, *Purposive Systems*, pages 15–23. Spartan Books, 1968.

[7] P. Fritsch. Five Mellow Guys Follow Their Dream: A 'Tall Ship' in Brazil. *The Wall Street Journal*, CXLII(35):1, Friday, February 18, 2000.

[8] Google. Google Home Page. Google, Inc. Mountain View, CA, June 2001.

[9] J.L. Gould and C.G. Gould. *The Honey Bee*. Scientific American Library, New York, 1988.

[10] S.G. Lee, T.U. Jung, and T.C. Chung. An Effective Dynamic Weighted Rule for Ant Colony System Optimization. In *Proceedings of CEC 2001*, pages 1393–1397. IEEE, Piscataway, NJ, 2001.

[11] M. Lindauer. *Communication Among Social Bees*. Harvard University Press, Cambridge, Massachusetts, 1961.

[12] F. Marckini. *Search Engine Positioning*. Wordware Publishing, Inc., Plano, TX, 2001.

[13] D. Merkle, M. Middendorf, and H. Schmeck. Ant Colony Optimization for Resource Constrained Project Scheduling. In *Proceedings of GECCO 2000*, pages 893–900. Morgan Kaufman Publishers, Inc., 2000.

[14] T. Ray and K.M. Liew. A Swarm with an Effective Information Sharing Mechanism for Unconstrained and Constrained Single Objective Optimization Problems. In *Proceedings of CEC 2001*, pages 75–80. IEEE, Piscataway, NJ, 2001.

[15] A. Vakali. A Web-based Evolutionary Model for Internet Data Caching. In *Proceedings of the* 10^{th} *International Workshop on Database and Expert Systems Applications*, pages 650–654. IEEE Press, 1999.

[16] R.L. Walker. Preliminary Study of Web Scouts/Foragers for a Bioinformatic Application: A Parallel Approach. In Y.V. Esteve, G.M. Carlomagno, and C.A. Brebbia, editors, *Computational Methods and Experimental Measurements X*, pages 967–976. WIT Press, 2001.

[17] R.L. Walker. Applying Evolutionary Computation Methodologies for Search Engine Development. In L. Wang, K.C. Tan, T. Furhashi, J. Kim, and X. Yao, editors, *SEAL'02: Proceedings of the 2002 Asia-Pacific Conference on Simulated Evolution and Learning*, pages 208–213, Singapore, November 2002. Nanyang Technological University Press.

[18] R.L. Walker. Using Nearest Neighbors to Discover Web Page Similarities. In H.R. Arabnia, editor, *PDPTA'02: Proceedings of the 2002 International Conference on Parallel and Distributed Processing Techniques and Applications*, pages 157–163. CSREA Press, June 2002.

[19] M. Wineberg and F. Oppacher. Enhancing the GA's Ability to Cope with Dynamic Environments. In *Proceedings of GECCO-2000*, pages 3–10. Morgan Kaufman Publishers, Inc., 2000.