ISSN 2395-8618

Editorial

WITH this issue I have yet another good news for you: in addition to ISI Thomson Web of Science (via the SciELO citation index), our journal is now indexed in Redalyc. The Redalyc scientific information system is a network of scientific journals of Latin America and the Caribbean, Spain and Portugal, whose mission is, according to their own statement, "to be a leading platform of open access scientific information services at international level, aimed at meeting the specialized information needs of students, researchers, and decision makers in the area of scientific and technological development, through retrieval and querying of specialized content and generation of metrics to assess quantitatively and qualitatively how science is being done in Ibero-America."

This new achievement is a result of the hard and high-quality work of the Associate Editors, editorial staff, reviewers, and most importantly the excellence of research contributed by our authors. Congratulations!

This issue of the journal Polibits includes ten papers by authors from nine different countries: Colombia, France, Hungary, Mexico, Oman, Saudi Arabia, Spain, Tunisia, and USA. The papers included in this issue are devoted to such topics as network modeling, image processing, data reduction, web analysis, natural language processing, organization modeling, web service design, enterprise content management, memetic algorithms, and job scheduling.

N. Meghanathan from **USA** in his paper "On the Sufficiency of Using Degree Sequence of the Vertices to Generate Random Networks Corresponding to Real-World Networks" investigates whether randomly generated networks with the same degree sequence as the modeled real-work network preserves important metrics of the original network. He shows that the properties of such randomly generated networks maintain very strong correlation with the original network on a number of node- and network-level metrics. This study enables the use of artificially generated networks to study the properties of real-world networks under different scenarios.

I. Hernández Bautista et al. from **Mexico** in their paper "Adjustment of Wavelet Filters for Image Compression Using Artificial Intelligence" present a method for lossless image compressing based on the lifting transform with automatic adjustment of wavelet filter coefficients. They show that their method achieves better compression. They use a pattern recognition method to optimize globally the parameters of the lifting filter for the image.

C. L. Sabharwal and **Bushra Anjum** from **USA** in their paper "Data Reduction and Regression Using Principal Component Analysis in Qualitative Spatial Reasoning and Health Informatics" show how to achieve data reduction in qualitative spatial reasoning and in health informatics, basing on principal component analysis along with traditional regression algorithms. On both artificial data and real data from the UCI Repository, they demonstrate that their method achieves both a better fit and more significantly reduced number of attributes than those produced by standard logistic regression.

C. Jebari from **Oman** in his paper "A Segment-based Weighting Technique for URL-based Genre Classification of Web Pages" proposes a segment-based weighting technique for genre classification of web pages. While the notion of genre is difficult to define precisely, it can be loosely understood as a category of artistic, musical, or literary composition that uses a particular style, form, or content. For computational analysis, there exist manually annotated datasets that allow quantifying an algorithm's ability to classify the data into genres. The method suggested in this paper achieves good results on three such datasets used for the experiments.

A. Novák from **Hungary** in his paper "Improving Corpus Annotation Quality Using Word Embedding Models" shows how to improve the quality of automatic annotation of web-crawled corpora using modern word embedding techniques. While automatically collecting and annotating text corpora for natural language processing saves huge effort and provides the researchers with low-cost very large annotated resources for training and testing machine learning methods, this simplicity has its cost in terms of the quality of the obtained resources. The author uses word embedding-based techniques to detect and correct or mitigate various types of language detection, encoding detection, and linguistic annotation errors. He experiments with the Hungarian-language corpora.

C. M. Zapata Jaramillo and **L. F. Castro Rojas** from **Colombia** in their paper "A Method Based on Patterns for Deriving Key Performance Indicators from Organizational Objectives" present a method for computational modeling organizations and assessing their performance indicators basing on such models. Their research is based on the technique proposed ten years ago by the first author, called pre-conceptual schema, which provides UML-like formal language for modeling interrelations between different elements in the organization. Using this method, the authors provide a systematic method for deriving a set of key performance indicators from a specific organizational objective. They illustrate their approach on the material of a specific case study.

G. Vargas-Solar et al. from **France** and **Spain** in their paper "Optimizing Data Processing Service Compositions Using SLA's" propose an approach for optimally accessing data by

ISSN 2395-8618

coordinating services according to Service Level Agreements for answering queries. They apply their approach to services that produce spatio-temporal data, in the scenario that lacks a full-fledged DBMS that would provide data management functions. Thus, the authors perform query evaluation using reliable service coordinations guided by Service Level Agreements, which are optimized for economic, energy, and time costs.

J. Márquez et al. from **Colombia** in their paper "Recommendation for an Enterprise Content Management (ECM) Based on Ontological Models" present an information retrieval system for an enterprise content manager, based on the use of ontologies. Their system gives the user the options to review the instances of the ontological model and to manage the aliases and ambiguities. The authors compare their system with traditional models.

E. Vega-Alvarado et al. from **Mexico** in their paper "A Memetic Algorithm Applied to the Optimal Design of a Planar Mechanism for Trajectory Tracking" describe a novel memetic algorithm that they called MABS, which is a modification of the Artificial Bee Colony Optimization, with a modified Random Walk algorithm for local search. They test their algorithm on one of optimization problems known to be particularly hard and at the same time important for industrial applications, namely, on the task of synthesis of a four-bar mechanism following a given trajectory. Simulation results show that the mechanism designed by their algorithm achieves high precision in following the desired trajectory, which shows that their algorithm can be successfully used for solving real-world practical tasks.

N. Nouri and **T. Ladhari** from **Tunisia** and **Saudi Arabia** in their paper "An Efficient Iterated Greedy Algorithm for the Makespan Blocking Flow Shop Scheduling Problem" propose an efficient algorithm for a kind of job scheduling problem called Blocking Flow Shop Scheduling Problem, which is characterized by the absence of buffer for waiting tasks. The algorithm they propose is of greedy type. It proceeds by making an adjustment between two relevant destruction and construction stages in order to minimize the maximum completion time. The performance of the algorithm is measured on Taillard's benchmark and compared with state-of-the-art methods.

This issue of the journal will be useful to researchers, students, and practitioners working in the corresponding areas, as well as to general public interested in advances in computer science, artificial intelligence, and computer engineering.

Dr. Alexander Gelbukh

Instituto Politécnico Nacional,
Mexico City, Mexico
Editor-in-Chief

ISSN 2395-8618

On the Sufficiency of Using Degree Sequence of the Vertices to Generate Random Networks Corresponding to Real-World Networks

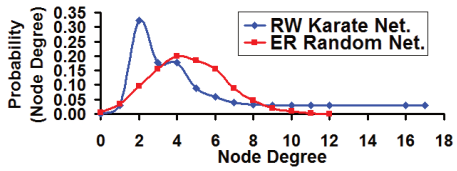
Natarajan Meghanathan

***Abstract*—The focus of research in this paper is to investigate whether a random network whose degree sequence of the vertices is the same as the degree sequence of the vertices in a real-world network would exhibit values for other analysis metrics similar to those of the real-world network. We use the well-known Configuration Model to generate a random network on the basis of the degree sequence of the vertices in a real-world network wherein the degree sequence need not be Poisson-style. The extent of similarity between the vertices of the random network and real-world network with respect to a particular metric is evaluated in the form of the correlation coefficient of the values of the vertices for the metric. We involve a total of 24 real-world networks in this study, with the spectral radius ratio for node degree (measure of variation in node degree) ranging from 1.04 to 3.0 (i.e., from random networks to scale-free networks). We consider a suite of seven node-level metrics and three network-level metrics for our analysis and identify the metrics for which the degree sequence would be just sufficient to generate random networks that have a very strong correlation (correlation coefficient of 0.8 or above) with that of the vertices in the corresponding real-world networks.**

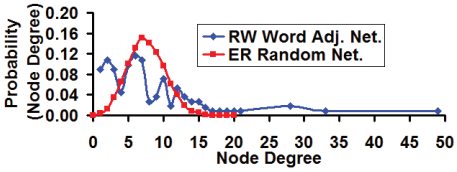
***Index Terms*—Configuration model, degree sequence, correlation, random network, real-world network.**

I. Introduction

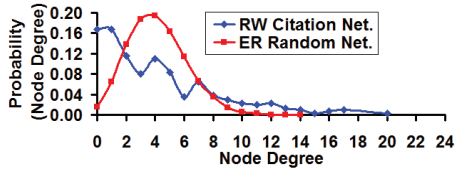
RANDOM networks are a class of complex networks in which there could be link between any two nodes in the network. The Erdos-Renyi (ER) model [1] is a commonly used theoretical model for generating random networks. The ER model-based random networks are characteristic of exhibiting a Poisson-style [2] degree sequence such that the degree of any vertex is typically very close to the average degree of the vertices in the network (i.e., the variation in node degree is typically low). However, the degree sequence of most of the real-world networks rarely follows a Poisson-style distribution; there usually exists an appreciable amount of variation in node degree [3] and there could also be preferential attachment with selected nodes rather than arbitrary attachment [4]. Figure 1 shows the degree sequence of the well-known real-world networks with N nodes and L edges and the corresponding ER model-based random networks generated with a probability of link value of $2L/\{N(N–1)\}$ [5]. Due to the inherent differences in the nature of the degree sequence, the values for several node-level metrics and network-level metrics exhibited by a ER model-based random network with a certain number of nodes and edges are likely to be independent (correlation coefficient is close to 0) to that of the metric values exhibited by a real-world network with the same number of nodes and edges [5].



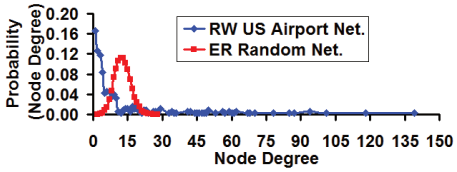
Karate Net. (34 nodes, 78 edges)



Word Adj. Net. (112 nodes, 425 edges)



Cit. Net. (311 nodes, 611 edges)



Airport Net. (332 nodes, 2126 edges)

Fig. 1. Degree Sequence of Real-World Networks and the Erdos-Renyi (ER) Model-based Random Networks with the Same Number of Nodes and Edges.

Manuscript received on January 30, 2016, accepted for publication on May 25, 2016, published on June 25, 2016.

Natarajan Meghanathan is with the Department of Computer Science at Jackson State University, MS 39217, USA (phone: 601-979-3661; fax: 601-979-2478; e-mail: natarajan.meghanathan@jsums.edu).

ISSN 2395-8618

In this research, we explore whether a random network whose degree sequence matches to that of a real-world network could exhibit similar values for several critical node-level and network-level metrics as that of the real-world network. In this pursuit, we chose to use the well-known Configuration Model [6] that takes as input the degree sequence of the vertices in any known network and generates a random network with a similar degree sequence. As can be seen in Section 5 of this paper, the correlation coefficient between the degrees of the vertices in the real-world network and the corresponding random network (generated using the Configuration Model) is 0.99 or above. We use a suite of seven node-level metrics and three network-level metrics to evaluate the similarity of each of the real-world network graphs and the corresponding Configuration Model-based random network graphs.

The node-level metrics analyzed are: Degree Centrality, Eigenvector Centrality [7], Betweenness Centrality [8], Closeness Centrality [9], Local Clustering Coefficient [10], Communicability [11] and Maximal Clique Size [12]; the network-level metrics analyzed are: Spectral Radius Ratio for Node Degree [3], Assortativity Index [13] and Algebraic Connectivity [5]. We use a total of 24 real-world network graphs (with different levels of variation in node degree) for this study. We run the appropriate algorithms to determine the individual node-level and network-level metrics on both the real-world graphs and the corresponding random graphs with identical degree sequence. We identify the levels of correlation based on the correlation coefficient values observed for each node-level metric in each of the real-world network graphs and their corresponding Configuration Model-based random network graphs as well as based on the percentage relative difference between the values for each network-level metric for the two graphs.

The rest of the paper is organized as follows: In Section 2, we review the Configuration Model for generating random networks and present a pseudo code for the implementation of the same. In Section 3, we introduce the node-level and network-level metrics evaluated in this paper and briefly describe the appropriate procedures to determine each of them. Section 4 reviews the Spearman's rank-based correlation measure [14] used in the analysis of the real-world networks. Section 5 introduces the real-world networks studied in this paper and analyzes the results for the levels of correlation for the node-level metrics and network-level metrics obtained on the real-world networks and the corresponding random networks generated using the Configuration Model based on the degree sequence of the real-world networks. Section 6 discusses related work on degree preserving randomization of real-world networks. Section 7 concludes the paper and outlines plans for future work. Throughout the paper, we use the terms 'node' and 'vertex', 'link' and 'edge', 'network' and 'graph' interchangeably. They mean the same.

II. Configuration Model

The Configuration Model [6] is one of the well-known models for generating random networks. Its unique characteristic is to take the degree sequence of a given network as input and generate a random network that has the same degree sequence as that of the input network. The degree sequence input to the model need not be Poisson-style - the typical pattern of degree sequence of vertices in random networks generated according to the well-known Erdos-Renyi (ER) model [1]. Thus, the Configuration Model could be used to generate random networks whose degree sequence could correspond to any network of analytical interest. In this paper, we use the Configuration Model to generate random networks whose degree sequence matches to that of real-world networks and we further evaluate the values of the node-level metrics and network-level metrics on both these networks. We are interested in exploring whether a random network whose degree sequence resembles to that of a real-world network exhibits similar values for other critical node-level and network-level metrics.

We simulate the generation of a random network under the Configuration Model as follows. Let N and L be respectively the total number of nodes and edges in a chosen real-world network. Let D be the set of degrees of the vertices (one entry per vertex) in the real-world network. We set up a list L_S of vertices - the number of entries for the ID of a vertex in this list is the degree of the vertex in the input set D. After the list L_S is constructed, we shuffle the entries in the list. We do the shuffling from the end of the list. In each iteration of shuffling, the ID of a vertex in a particular entry in the list at index i ($|L_S| \geq i \geq 2$) is swapped with the ID of a vertex in a randomly chosen entry at index j ($j < i$). We now generate the adjacency matrix A_{conf} (each entry is initialized to zero) for the configured graph as follows. We consider the vertex IDs from the end of the shuffled list L_S. For each vertex *uID* at index u ($|L_S| \geq u \geq 2$) considered, we attempt to pair it with a vertex *vID* at index v ($v < u$) such that $A_{conf}[uID][vID] = 0$ and *uID* is not the same as *vID* as well as make sure the entry at index v has not been already paired with another vertex. To keep track of the latter, we set the entries of the shuffled list L_S to –1 if the entry is already considered either as an *uID* or a *vID*. If a pair (*uID*, *vID*) meets the above criteria, we set the entries $A_{conf}[uID][vID] = 1$ and $A_{conf}[uID][vID] = 1$. We proceed the iterations until the index u equals 1; by this time, all entries in the shuffled list L_S should have been set to –1.

The above implementation procedure for the Configuration model does not generate any self-loop or duplicate edge, as we make sure we are not pairing a vertex with a particular ID at an index u to a vertex with the same ID at another index v as well as we keep track of the edges that have been already configured across all the iterations. To test whether a pair (*uID*, *vID*) already have an edge between them, we have to just check the entries for *uID* or *vID* in A_{conf}. Thus, each

ISSN 2395-8618

```
Input: Degree sequence D of the Vertices; Number of Nodes, N
Output: Adjacency Matrix of the Configured Graph A_conf[1...N][1...N]
Auxiliary Variables: List L_S; Total Entries
Initialization: A_conf[uID, vID] = 0, where 1 ≤ uID ≤ N and 1 ≤ vID ≤ N; Total Entries = 0
Begin Generate Graph-Configuration Model
1   for 1 ≤ uID ≤ N do
2      L_S[Total Entries + 1 ... D[uID]] = uID
3      Total Entries = Total Entries + D[uID]
4   end for
5   for (i = Total Entries; i > 1; i = i - 1) do
6      Generate a random index j ∈ {i-1...1}
7      Swap(L_S[i], L_S[j])
8   end for
9   for (u = Total Entries; u > 1; u = u-1) do
10     uID = L_S[u]
11     if (uID != -1) then
12              Generate a random index v ∈ {u-1...1} and find a vID = L_S[v]
                        such that uID != vID and L_S[vID] != -1 and A_conf[uID][vID] = 0
13             L_S[uID] = -1
14             L_S[vID] = -1
15             A_conf[uID][vID] = 1
16             A_conf[vID][uID] = 1
17     end if
18  end for
return A_conf
End Generate Graph-Configuration Model
```

Fig. 2. Pseudo Code for the Implementation of the Configuration Model to Generate Random Graph according to a given Degree sequence.

iteration (lines 9–17) is likely to take at most O(N) attempts before a link (*uID*, *vID*) is configured. The total number of iterations involving lines 5-8 and lines 9-17 is the sum of the degrees of the vertices in the chosen real-world network. Note that the sum of the degrees of the vertices in a graph is equal to $2L/N$ where L is the number of links and N is the number of nodes. Hence, the overall time-complexity of the implementation of the Configuration Model described in Figure 2 is O($N \times 2L/N$) = O(L).

Figure 3 presents an example to illustrate the generation of a random graph that has the same degree sequence as that of an input graph. The example walks through the sequence of iterations illustrating the execution of the pseudo code given in Figure 2.

We show the contents of the list L_S at the time of initialization (before and after shuffling) as well as during each iteration (before and after the configuration of an edge).

Whenever a vertex pair is picked up for configuring an edge, we replace their entries with –1. We also show sample scenarios wherein we reject the choice of a *vID* if it is same as that of the *uID* (shown with a ✕ in iterations 3 and 6) as well as show a sample scenario wherein we reject the choice of a *vID* (shown with a ✕ in iteration 5) to avoid adding a duplicate edge for the pair (*uID*, *vID*). The final configured graph has exactly 8 edges and degree sequence as that of the input graph. However, the edges in the input graph do not match to that of the edges in the final configured graph. As a result, it is not clear whether several other node-level metrics (like the centrality measures, clustering coefficient, communicability, etc) and network-level metrics (like edge assortativity, algebraic connectivity) would be the same for the two graphs. This is the motivation for the research conducted in the rest of the paper.

III. Node-Level Metrics and Network-Level Metrics

Our objective in this paper is to identify the node-level metrics and network-level metrics for which the degree sequence would be sufficient to observe a very strong correlation between a chosen real-world network graph and its corresponding configuration model generated random network graph. In this pursuit, we study the following node-level metrics (eigenvector centrality, closeness centrality, betweenness centrality, local clustering coefficient, communicability and maximal clique size) and network-level metrics (spectral radius ratio for node degree, edge assortativity and algebraic connectivity).

A. Eigenvector Centrality

The eigenvector centrality (EVC) of a vertex is a measure of the degree of the vertex as well as the degree of its neighbors. The EVC of the vertices in a graph is obtained by computing the principal eigenvector of the adjacency matrix (A) of the graph. In this paper, we use the Power-Iteration

ISSN 2395-8618

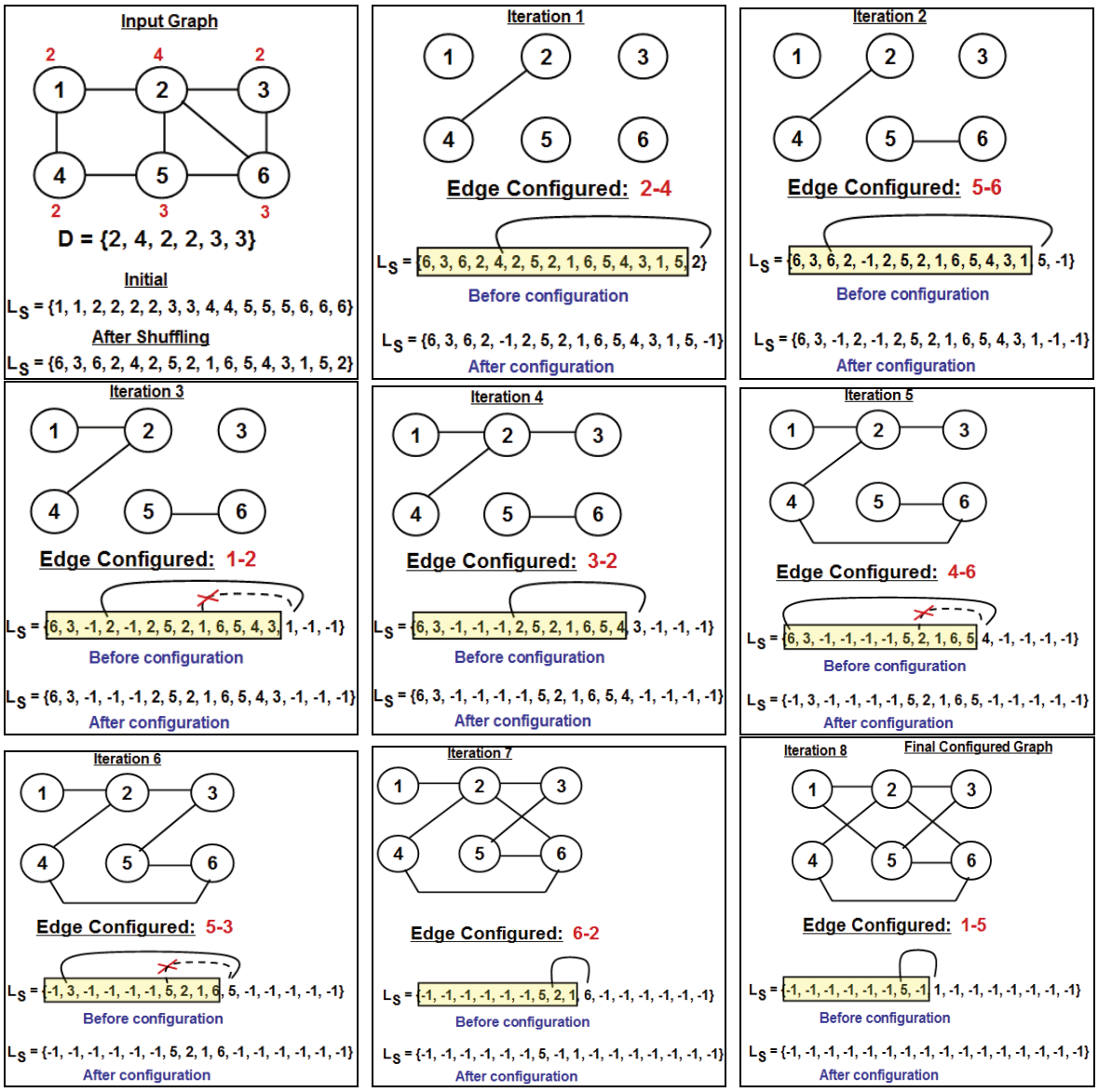


Fig. 3. Example Execution of the Implementation of the Configuration Model to Generate Random Graph according to a given Degree sequence.

algorithm [15] to determine the principal eigenvector/EVC of the vertices. This algorithm is briefly explained as follows.

We start with a unit-column vector of all 1s X_0 = [1 1 1 1 ... 1] as the estimated principal eigenvector of the graph where the number of 1s is the number of vertices in the graph. In the $(i+1)^{th}$ iteration, the principal eigenvector $X_{i+1} = AX_i / \|AX_i\|$ where $\|...\|$ is the normalized value of the product vector obtained by multiplying the adjacency matrix A and the column vector X_i.

We continue the iterations until the normalized value of the product vector (as indicated above) does not change beyond a certain level of precision for subsequent iterations. There exists an entry for each of the vertices in the principal eigenvector and the values in these entries correspond to the EVC of the vertices.

Figure 4 illustrates an example to compute the eigenvector centrality of the vertices in a graph using the Power-iteration algorithm. We stop when the normalized value (in the example, it is 2.85) of the product of the adjacency matrix and the principal eigenvector converges and does not change beyond the second decimal. Vertex 2 has the highest EVC followed by vertex 6. We notice that though the three vertices 1, 3 and 4 have the same degree, they differ in their EVC values: Both the neighbors of Vertex 3 are vertices with higher EVC - as a result, the EVC of vertex 3 is relatively higher than that of vertices 1 and 4. Vertex 1 has a higher EVC than vertex 4 (vertices 1 and 4 are also connected to each other) because vertex 1 is connected to a vertex with a higher EVC (vertex 2) while vertex 4 is connected to a vertex (vertex 5) with a relatively lower EVC.

ISSN 2395-8618

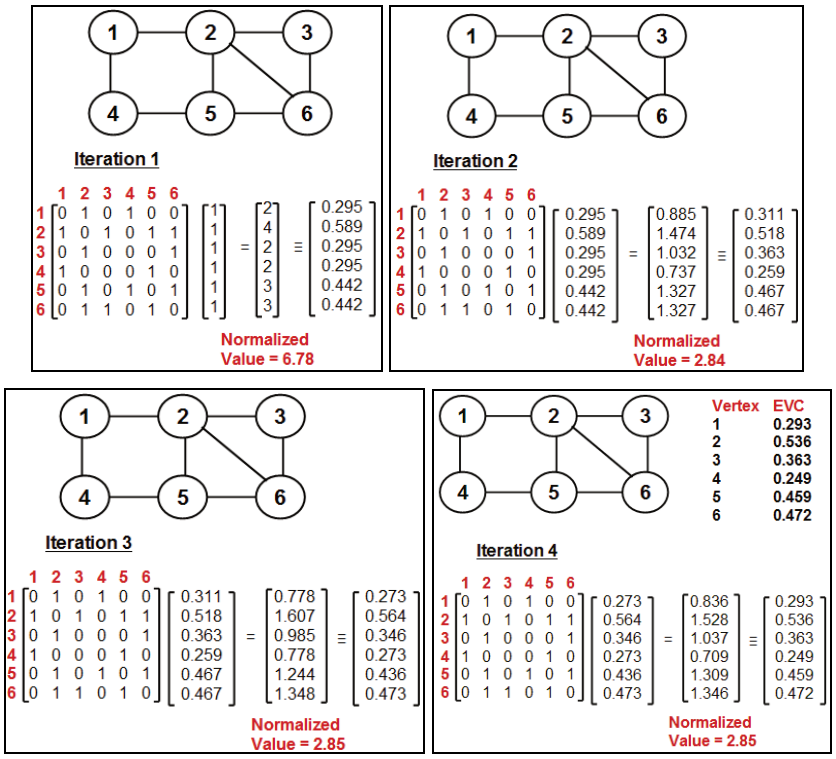


Fig. 4. Example to Illustrate the Execution of the Power-Iteration Algorithm to Determine the Eigenvector Centrality of the Vertices in a Graph.

B. *Closeness Centrality*

The closeness centrality (ClC) [9] of a vertex is the inverse of the sum of the shortest path distances (number of hops) from the vertex to the rest of the vertices in the graph. The ClC of a vertex is determined by running the Breadth First Search (BFS) algorithm [16] on the vertex and determining a shortest path tree rooted at the vertex. One can then easily determine the sum of the number of hops from the vertex to the other vertices in the shortest path tree and the closeness centrality of the vertex is the inverse of this sum. Figure 5 illustrates an example to compute the closeness centrality of the vertices in a graph. We show the shortest path trees rooted at each vertex and compute the number of hops from the root to the rest of the vertices in these trees to arrive at the distance matrix, contributing to the computation of the closeness centrality.

C. *Betweenness Centrality*

The betweenness centrality (BWC) [8] of a vertex is a measure of the fraction of the shortest paths between any two vertices that go through the particular vertex, summed over all pairs of vertices. The number of hops for a vertex from the root of a shortest path tree indicates the level of the vertex on the tree. The number of shortest paths, denoted sp_{jk}, from a vertex j to a vertex k at level l ($l > 0$) is the sum of the number of shortest paths from j to each of the neighbors of k (in the original graph) that are at level l-1 in the shortest path tree rooted at j. For any vertex i, the number of shortest paths from vertex j to vertex k that go through i, denoted $sp_{jk}(i)$, is the maximum of the number of shortest paths from vertex j to vertex i and the number of shortest paths from vertex k to vertex i. Quantitatively, the BWC of a vertex i is defined as

$$BWC(i) = \sum_{j \neq k \neq i} \frac{sp_{jk}(i)}{sp_{jk}}.$$

Figure 6 shows an example illustrating the computation of the BWC of the vertices in a graph.

D. *Local Clustering Coefficient*

The local clustering coefficient (LCC) [10] of a vertex in a graph is a measure of the probability that any two neighbors of the vertex are connected. Quantitatively, the local clustering coefficient of a vertex is the ratio of the actual number of links between the neighbors of the vertex divided by the maximum possible number of links between the neighbors of the vertex. For a vertex i with degree k_i, if there are a total of l links connecting the neighbors of i, then the clustering coefficient of i is

$$\frac{l}{k_i(k_i - 1)/2}.$$

Figure 7 shows an example of computing the local clustering coefficient of the vertices of a graph.

ISSN 2395-8618

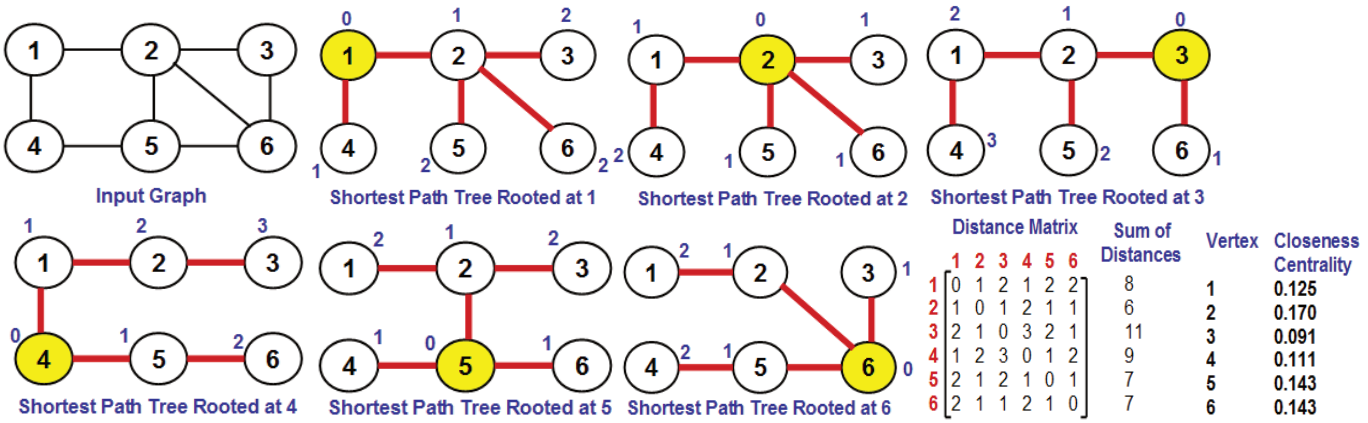
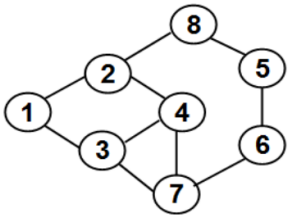


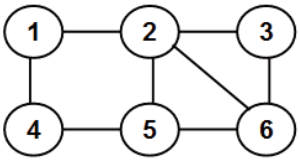
Fig. 5. Example to Illustrate the Computation of the Closeness Centrality of the Vertices.




```
betweeness for node: 1
Pair (3,2) ---> 1 / 2
Pair (2,3) ---> 1 / 2
Pair (8,3) ---> 1 / 2
Pair (3,8) ---> 1 / 2
betweeness for node: 2
Pair (4,1) ---> 1 / 2
Pair (5,1) ---> 1 / 1
Pair (8,1) ---> 1 / 1
Pair (8,3) ---> 2 / 2
Pair (1,4) ---> 1 / 2
Pair (5,4) ---> 1 / 2
Pair (8,4) ---> 1 / 1
Pair (1,5) ---> 1 / 1
Pair (4,5) ---> 1 / 2
Pair (8,7) ---> 1 / 2
Pair (1,8) ---> 1 / 1
Pair (3,8) ---> 2 / 2
Pair (4,8) ---> 1 / 1
Pair (7,8) ---> 1 / 2
betweeness for node: 3
Pair (4,1) ---> 1 / 2
Pair (6,1) ---> 1 / 1
Pair (7,1) ---> 1 / 1
Pair (1,4) ---> 1 / 2
Pair (1,6) ---> 1 / 1
Pair (1,7) ---> 1 / 1
betweeness for node: 4
Pair (3,2) ---> 1 / 2
Pair (6,2) ---> 1 / 2
Pair (7,2) ---> 1 / 1
Pair (2,3) ---> 1 / 2
Pair (8,3) ---> 1 / 2
Pair (2,6) ---> 1 / 2
Pair (2,7) ---> 1 / 1
Pair (8,7) ---> 1 / 2
Pair (3,8) ---> 1 / 2
Pair (7,8) ---> 1 / 2
```

```
betweeness for node: 5
Pair (6,2) ---> 1 / 2
Pair (2,6) ---> 1 / 2
Pair (8,6) ---> 1 / 1
Pair (8,7) ---> 1 / 2
Pair (6,8) ---> 1 / 1
Pair (7,8) ---> 1 / 2
betweeness for node: 6
Pair (5,3) ---> 1 / 1
Pair (5,4) ---> 1 / 2
Pair (3,5) ---> 1 / 1
Pair (4,5) ---> 1 / 2
Pair (7,5) ---> 1 / 1
Pair (5,7) ---> 1 / 1
Pair (8,7) ---> 1 / 2
Pair (7,8) ---> 1 / 2
betweeness for node: 7
Pair (6,1) ---> 1 / 1
Pair (6,2) ---> 1 / 2
Pair (5,3) ---> 1 / 1
Pair (6,3) ---> 1 / 1
Pair (5,4) ---> 1 / 2
Pair (6,4) ---> 1 / 1
Pair (3,5) ---> 1 / 1
Pair (4,5) ---> 1 / 2
Pair (1,6) ---> 1 / 1
Pair (2,6) ---> 1 / 2
Pair (3,6) ---> 1 / 1
Pair (4,6) ---> 1 / 1
betweeness for node: 8
Pair (5,1) ---> 1 / 1
Pair (5,2) ---> 1 / 1
Pair (6,2) ---> 1 / 2
Pair (5,4) ---> 1 / 2
Pair (1,5) ---> 1 / 1
Pair (2,5) ---> 1 / 1
Pair (4,5) ---> 1 / 2
Pair (2,6) ---> 1 / 2
```

Fig. 6. Example to Illustrate the Computation of the Betweenness Centrality of the Vertices.



Vertex	Neighbors	# Links connecting the Neighbors	Max. possible # Links connecting the Neighbors	Local Clustering Coefficient
1	2, 4	0	2(1)/2 = 1	0/1 = 0.0
2	1, 3, 5, 6	2	4(3)/2 = 6	2/6 = 0.33
3	2, 6	1	2(1)/2 = 1	1/1 = 1.0
4	1, 5	0	2(1)/2 = 1	0/1 = 0.0
5	2, 4, 6	1	3(2)/2 = 3	1/3 = 0.33
6	2, 3, 5	2	3(2)/2 = 3	2/3 = 0.67

Fig. 7. Example to Illustrate the Computation of the Clustering Coefficient of the Vertices.

E. Communicability

The communicability (COMM) [11] of a vertex is the weighted sum of the number of walks of lengths l = 1, 2, 3, ... from that vertex to each of the other vertices in the graph, with the weight being 1/l!. A walk from vertex r to s involves a sequence of intermediate vertices that may or may not appear more than once. That is, a walk could involve cycles. The communicability of a vertex captures the ease with which a vertex can disseminate information to the rest of the vertices through various walks (the shortest paths are given more weights though). Though the definition of the communicability of a vertex could be represented mathematically as in equation (1), we use the closed form equation (2) to quantitatively determine the communicability of the vertices in a graph [11]. $(A^l)_{rs}$ represents the number of walks of length l between two vertices r and s. Note that for a graph of n vertices (where V is the set of vertices, $|V| = n$), there are n eigenvalues (denoted as λ_j where j = 1, 2, ..., n) and the corresponding eigenvectors (denoted as ϕ_j, where j = 1, 2, ..., n). $\phi_j(r)$ and $\phi_j(s)$ denotes the values for vertices r and s in the eigenvector associated with eigenvalue λ_j. We compute the

ISSN 2395-8618

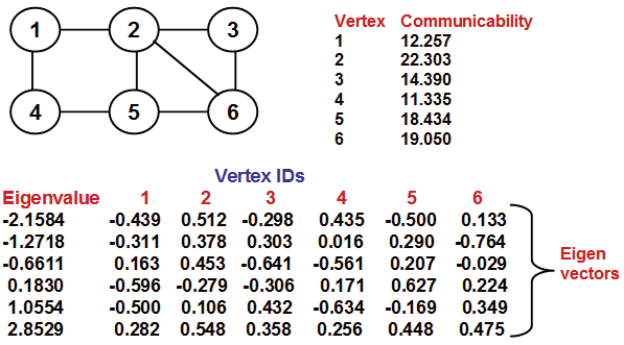


Fig. 8. Example to Illustrate the Computation of the Clustering Coefficient of the Vertices.

eigenvalues and eigenvectors of the adjacency matrix of a graph using the JAMA package [17].

$$C(r) = \sum_{s \in V-\{r\}} \sum_{l=1}^{\infty} \frac{(A^l)_{rs}}{l!} \quad (1)$$

$$C(r) = \sum_{s \in V-\{r\}} \sum_{j=1}^{n} \varphi_j(r)\varphi_j(s)e^{\lambda_j} \quad (2)$$

F. *Maximal Clique Size*

A clique is a subset of the vertices of a graph such that there exists an edge between any two vertices in this set. Each vertex in a graph is part of at least one clique, as even an edge could be considered a clique of size 2. We refer to the maximal clique for a vertex as the largest size clique that the vertex is part of and call the size of the corresponding clique as the maximal clique size [12]. We refer to the maximum clique size of the entire graph as the largest of the maximal clique size (MCS) values of the vertices [18]. As observed in the example shown in Figure 9, one or more vertices (vertices 4, 5, 6, 7) could be part of a maximum clique size, while for the rest of the vertices (vertices 1, 2 and 3), the maximal clique size could be less than maximum clique size. We use the extended version of an exact algorithm by Pattabiraman et al [18] to determine the maximal clique size for each vertex. The algorithm takes a branch and bound approach of exploring all possible candidate cliques that a vertex could be part of, but searching through only viable candidate sets of vertices whose agglomeration has scope of being a clique of size larger than the currently known clique found as part of the search.

Figure 8 presents the communicability of the vertices for the same example graph (of six vertices) used in Figures 4, 5 and 7. The figure also lists the six eigenvalues and the corresponding eigenvectors that are used in the calculations of the communicability of the vertices (according to equation 2). We observe vertex 2, followed by vertex 6, to have the largest values for communicability. In general, vertices having a higher degree and part of a closely-knit community (vertices

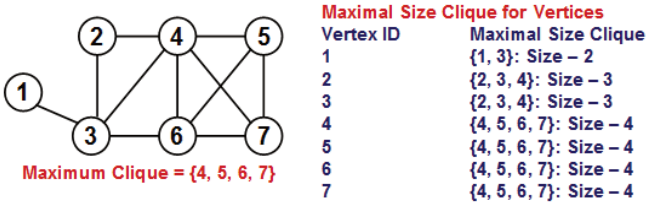


Fig. 9. Example to Illustrate the Maximal Size Clique of the Vertices.

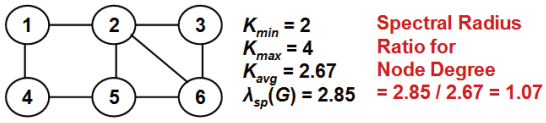


Fig. 10. Example to Illustrate the Relationship between Spectral Radius and Node Degree.

2, 3, 5 and 6 would have formed a clique had there been an edge 3-5). Notice that between vertices 5 and 6 (both of which have degree 3), vertex 6 has a slightly larger communicability, attributed to the connection of vertex 6 to vertex 3 that is in turn connected to vertex 2 (whereas vertex 5 is connected to vertex 4 that is not connected to vertex 2, but instead connected to a low-degree vertex, vertex 1). Likewise, between vertices 1, 3 and 4 (all of which have degree 2), vertex 3 has the highest communicability as it is connected to vertices 2 and 6 - both of which have a high communicability.

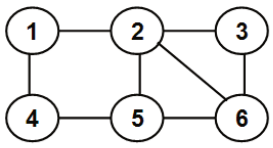
G. *Spectral Radius Ratio for Node Degree*

The spectral radius of a graph G, denoted $\lambda_{sp}(G)$, is the principal eigenvalue (largest eigenvalue) of the adjacency matrix of the graph. If k_{min}, k_{avg} and k_{max} are the minimum, average and maximum node degrees, then $k_{min} \leq k_{avg} \leq \lambda_{sp}(G) \leq k_{max}$ [19]. As one can see from this relationship, the spectral radius could be construed as a measure of the variation in the degree of the vertices in a graph. In [3], the notion of spectral radius ratio for node degree was proposed to evaluate the variation in node degree on a uniform scale, without the need for explicitly computing the variance/standard deviation of the vertices in the graph. The spectral radius ratio for node degree is the ratio of the spectral radius of the graph and the average degree of the vertices in the graph: $\lambda_{sp}(G)/k_{avg}$. According to the above formulation, the spectral radius ratio for node degree values are always 1.0 or above; the farther the value is from 1.0, the larger the variation in node degree among the vertices of the graph. Figure 10 presents an example to illustrate the relationship $k_{min} \leq k_{avg} \leq \lambda_{sp}(G) \leq k_{max}$ and the spectral radius ratio for node degree. As this ratio is closer to 1.0, we could construe that the variation in node degree is very less; we can see 50% (three out of six) of the vertices have degree 2 and one-third (two out of six) of the vertices have degree 3, leading to an average degree of 2.67.

ISSN 2395-8618

H. Edge Assortativity

The assortativity of the edges in a graph is a measure of the similarity of the end vertices of the edges based on any notion of node weights [13]. In this research, we use node degree as the measure of node weight. Quantitatively, edge assortativity is essentially the correlation coefficient of the node weights of the end vertices. If the correlation coefficient is close to 1.0, -1.0 and 0.0 respectively, we could say the end vertices of the edges are respectively maximally similar, maximally different and independent to each other based on the notion of node weights considered. Figure 11 presents an example to calculate edge assortativity in a graph, wherein the ids of the vertices constituting an edge are considered as an ordered pair (i, j) such that $i < j$. We observe the correlation coefficient (edge assortativity) to be close to 0.0, indicating that the pairing of the vertices that constitute the edges of the graph is independent of the degrees of the end vertices constituting these edges. ER model-based Random graphs [1] exhibit an edge assortativity close to 0 indicating the arbitrary pairing of the vertices to constitute the edges.



$$\text{Edge Assortativity} = \frac{\sum_{i=1}^{n}(Xi-\bar{X})(Yi-\bar{Y})}{\sum_{i=1}^{n}(Xi-\bar{X})^2 \sum_{i=1}^{n}(Yi-\bar{Y})^2}$$

Edge	Vertex Weights X	Y	(Xi – X̄)	(Yi – Ȳ)	(Xi-X̄)^2	(Yi-Ȳ)^2	(Xi-X̄)(Yi-Ȳ)
1 – 2	2	4	-0.88	1.12	0.7744	1.2544	-0.9856
1 – 4	2	2	-0.88	-0.88	0.7744	0.7744	0.7744
2 – 3	4	2	1.12	-0.88	1.2544	0.7744	-0.9856
2 – 5	4	3	1.12	0.12	1.2544	0.0144	1.2544
2 – 6	4	3	1.12	0.12	1.2544	0.0144	0.1344
3 – 6	2	3	-0.88	0.12	0.7744	0.0144	-0.1056
4 – 5	2	3	-0.88	0.12	0.7744	0.0144	-0.1056
5 – 6	3	3	0.12	0.12	0.0144	0.0144	0.0144
Average	2.88	2.88		Sum	6.8752	2.8752	-0.0048
	X̄	Ȳ					

Edge Assortativity = -0.000248

Fig. 11. Example to Illustrate the Calculation of Edge Assortativity as a Correlation Coefficient of the Node Weights of the End Vertices.

I. Algebraic Connectivity

The algebraic connectivity of a graph is a quantitative measure of the connectivity of the graph capturing the vulnerability of a graph for disconnection as a function of the number of vertices in the graph as well as the topology of the graph [20]. The algebraic connectivity of a graph is bounded above by the traditional connectivity of the graph, defined as the minimum number of vertices that need to be removed to disconnect the graph into two or more components [21]. However, the traditional connectivity measure (an integer corresponding to the minimum number of vertices to be removed for disconnection) cannot capture the relative strength of the graph with respect to node removals. For two graphs having the same value of traditional connectivity, the algebraic connectivity could be still different [21]. The larger the value of the algebraic connectivity, the stronger the graph – only the removal of certain nodes could disconnect the graph (and not the removal of any node).

Quantitatively, for a connected graph, the algebraic connectivity is measured to be the second smallest eigenvalue of the Laplacian Matrix (L) of a graph [22]. In addition, for a connected graph, the smallest eigenvalue of the Laplacian Matrix of the graph is always 0. In general, the number of zeros among the eigenvalues of the Laplacian Matrix of a graph indicates the number of connected components of the graph [23]. The entries in the Laplacian Matrix of a graph are defined as follows [23]:

$L(i, j) = \text{degree}(i)$ if $i = j$
$= -1$ for $i \neq j$ and edge (i, j) exists
$= 0$ for $i \neq j$ and edge (i, j) does not exist

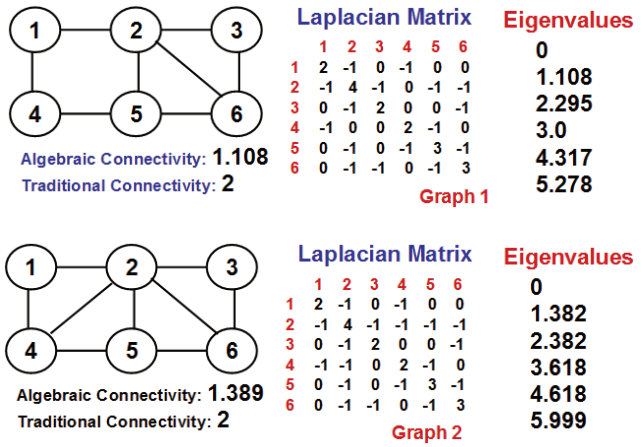


Fig. 12. Example to Illustrate the Determination of Algebraic Connectivity and its Use a Measure of Evaluation of the Relative Strengths of Two Graphs with the Same Traditional Connectivity.

Figure 12 presents examples to determine the Laplacian Matrices of two graphs and compute the sequence of eigenvalues for the two matrices. There is only one zero among the eigenvalues of the Laplacian matrices of both the graphs, indicating that both the graphs are connected and all the vertices form a single connected component. Graph-2 is relatively stronger than Graph-1 due to the presence of an additional edge 2-4 in the former.

Though both the graphs have a traditional connectivity of 2 (both the graphs get disconnected with the removal of vertices 2 and 5 in each of them), the removal of any two vertices from the graph is relatively more likely to lead to a disconnection in Graph-1 compared to Graph-2. One could notice that the removal of vertices 1 and 5 could disconnect vertex 4 from the rest of the vertices in Graph-1; on the other hand, the removal of vertices 1 and 5 would not disconnect vertex 4 from the rest of the vertices in Graph-2.

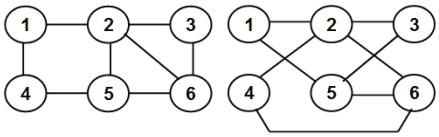
ISSN 2395-8618

IV. Spearman's Rank-based Correlation Measure

We resort to a rank-based correlation coefficient study in this paper as we want to explore the level of similarity between the ranking of the vertices (with respect to a node-level metric) in a real-world network graph and the corresponding degree-preserved random network graph. In this pursuit, we choose to use the Spearman's rank-based correlation measure (SCC). SCC is a measure of how well the relationship between two datasets (variables) can be assessed using a monotonic function [14]. To compute the SCC of two datasets (say, A and C), we convert the raw scores A_i and C_i for a vertex i to ranks a_i and c_i and use formula (3) shown below, where $d_i = a_i - c_i$ is the difference between the ranks of vertex i in the two datasets. We follow the convention of assigning the rank values from 1 to n for a graph of n vertices with vertex IDs that are also assumed to range from 1 to n. To obtain the rank for a vertex based on the list of values for a node-level metric, we first sort the values (in ascending order). If there is any tie, we break the tie in favor of the vertex with a lower ID; we will thus be able to arrive at a tentative, but unique, rank value for each vertex with respect to the metric. We determine a final ranking of the vertices as follows: For vertices with unique value of the node-level metric, the final ranking is the same as the tentative ranking. For vertices with an identical value for the node-level metric, the final ranking is assigned to be the average of their tentative rankings. Figure 13 illustrates the computation of the tentative and final ranking of the vertices based on their BWC values in the actual graph and the configuration model-based random network graph generated in Figure 3 as well as illustrates the computation of the Spearman's rank-based correlation coefficient.

$$SCC(A,C) = 1 - \frac{6\sum_{i=1}^{n} d_i^{\,2}}{n(n^2-1)} \qquad (3)$$

In Figure 13, we observe ties among vertices with respect to BWC in both the actual graph and the corresponding configured graph. The tentative ranking is obtained by breaking the ties in favor of vertices with lower IDs. In the case of the actual graph, we observe both vertices 1 and 6 to have an identical BWC value of 0.83 each and their tentative rankings are respectively 3 and 4 (ties for tentative rankings are broken in favor of vertices with lower IDs); the final ranking (3.5) for both these vertices is thus the average of 3 and 4. A similar scenario could be observed for the configured graph: vertices 1 and 3 have an identical BWC value of 0.33 each and their tentative rankings are respectively 2 and 3; the final ranking (2.5) for both these vertices is thus the average of 2 and 3. The Spearman's rank-based correlation coefficient with respect to BWC for the actual graph and configured graph in Figure 13 is observed to be 0.87, indicating a very strong positive correlation. A ranking of the vertices with respect to BWC in the actual graph is: 3, 4, 1-6 (tie), 5 and 2; whereas the ranking of the vertices with respect to BWC in the configured graph is: 4, 1-3 (tie), 6, 5 and 2.



Actual Graph	
Vertex ID	BWC
1	0.83
2	3.17
3	0.00
4	0.50
5	2.17
6	0.83

Configured Graph	
Vertex ID	BWC
1	0.33
2	3.50
3	0.33
4	0.00
5	1.50
6	1.33

Vertex	BWC in Actual Graph [A]	Tentative Rank: A	Final Rank: a_i	BWC in Config Graph [C]	Tentative Rank: C	Final Rank: c_i	Rank Difference (d_i): $a_i - c_i$	d_i^2
1	0.83	3	3.5	0.33	2	2.5	1	1
2	3.17	6	6	3.50	6	6	0	0
3	0.00	1	1	0.33	3	2.5	-1.5	2.25
4	0.50	2	2	0.00	1	1	1	1
5	2.17	5	5	1.50	5	5	0	0
6	0.83	4	3.5	1.33	4	4	-0.5	0.25
							Sum	4.5

Spearman's Rank Correlation Coefficient = 1 - (6*4.5)/(6*(62-1)) = 0.87

Fig. 13. Example to Illustrate the Computation of the Spearman's Rank-based Correlation Coefficient with respect to BWC on the Actual Graph and Configured Graph of Figure 3.

The correlation coefficient values obtained for all the node-level metrics range from -1 to 1. Correlation coefficient values closer to 1 for a node-level metric indicate that identical degree sequence for the real-world network graph and the configuration model based random network graph is sufficient to generate an identical ranking of the vertices in the two graphs with respect to the metric. Correlation coefficient values closer to -1 for a node-level metric indicate that identical degree sequence between the real-world network graph and the configuration model based random network graph is sufficient to generate a ranking of the vertices in the real-world network graph that is the reversal of the ranking of the vertices in the corresponding configuration model-based random network graph (i.e., a highly ranked vertex with respect to the particular node-level metric in the real-world network graph is ranked much low with respect to the same metric in the corresponding random network graph and vice-versa). Correlation coefficient values closer to 0 indicate no correlation (i.e., an identical degree sequence alone is not sufficient to generate an identical ranking of the vertices with respect to the node-level metric in the real-world network graph and the corresponding random network graphs). We will adopt the ranges (rounded to two decimals) proposed by Evans [23] to indicate the various levels of correlation, shown in Table 1. The color code to be used for the various levels of correlation is also shown in this table.

ISSN 2395-8618

TABLE I
RANGE OF CORRELATION COEFFICIENT VALUES AND THE CORRESPONDING LEVELS OF CORRELATION

Range of Correlation Coefficient Values	Level of Correlation
0.80 to 1.00	Very Strong Positive
0.60 to 0.79	Strong Positive
0.40 to 0.59	Moderate Positive
0.20 to 0.39	Weak Positive
0.01 to 0.19	Very Weak Positive
0.00	Neutral
–0.19 to –0.01	Very Weak Negative
–0.39 to –0.20	Weak Negative
–0.59 to –0.40	Moderate Negative
–0.79 to –0.60	Strong Negative
–1.00 to –0.80	Very Strong Negative

V. REAL-WORLD NETWORKS

We analyze a total of 24 real-world networks with different levels of variation in node degree. The spectral radius ratio for node degree ($\lambda_{sp}(k)$) for these networks varies from 1.04 to 3.0, with the type of networks ranging from random networks to scale-free networks [4]. All networks are modeled as undirected networks. A brief description of the 24 real-world networks (a three-character abbreviation for each of these networks is indicated in the parenthesis), in the increasing order of their spectral radius ratio for node degree, is as follows:

(1) Macaque Dominance Network (MDN) [24]: This is a network of 62 adult female Japanese macaques (monkeys; vertices) in a colony, known as the "Arashiyama B Group", recorded during the non-mating season from April to early October 1976. There exists an edge between two vertices if the one of the two corresponding macaques exhibited dominance over the other macaque.

(2) College Fraternity Network (CFN) [25]: This is a network of 58 residents (vertices) in a fraternity at a West Virginia college; there exists an edge between two vertices if the corresponding residents were seen in a conversation at least once during a five day observation period.

(3) Hypertext 2009 Network (HTN) [26]: This is a network of the face-to-face contacts of 113 attendees (vertices) of the ACM Hypertext 2009 conference held in Turin, Italy from June 29 to July 1, 2009. There exists an edge between two vertices if the corresponding conference visitors had face-to-face contact that was active for at least 20 seconds.

(4) Flying Teams Cadet Network (FTC) [27]: This is a network of 48 cadet pilots (vertices) at an US Army Air Forces flying school in 1943 and the cadets were trained in a two-seated aircraft. There exists an edge between two vertices if at least one of the two corresponding pilots has indicated the other pilot as his/her preferred partner with whom s/he likes to fly during the training schedules.

(5) Sawmill Strike Communication Network (SSC) [28]: This is a network of 24 employees (vertices) in a sawmill who planned a strike against the new compensation package proposed by their management. There exists an edge between any two vertices if the corresponding employees mutually admitted (to an outside consultant) discussing about the strike with a frequency of three or more (on a 5-point scale).

(6) Primary School Contact Network (PSN) [29]: This is a network of children and teachers (vertices) used in the study published by an article in BMC Infectious Diseases, 2014. There exists an edge between two vertices if the corresponding persons were in contact for at least 20 seconds during the observation period.

(7) Mexican Political Elite Network (MPN) [30]: This is a network of 35 Mexican presidents and their close collaborators (vertices); there exists an edge between two vertices if the corresponding two people have ties that could be either political, kinship, friendship or business ties.

(8) Residence Hall Friendship Network (RHF) [31]: This is a network of 217 residents (vertices) living at a residence hall located on the Australian National University campus. There exists an edge between two vertices if the corresponding residents are friends of each other.

(9) UK Faculty Friendship Network (UKF) [32]: This is a network of 81 faculty (vertices) at a UK university. There exists an edge between two vertices if the corresponding faculty are friends of each other.

(10) World Trade Metal Network (WTM) [33]: This is a network of 80 countries (vertices) that are involved in trading miscellaneous metals during the period from 1965 to 1980. There exists an edge between two vertices if one of the two corresponding countries imported miscellaneous metals from the other country.

(11) Jazz Band Network (JBN) [34]: This is a network of 198 Jazz bands (vertices) that recorded between the years 1912 and 1940; there exists an edge between two vertices if the corresponding bands had shared at least one musician in any of their recordings during this period.

(12) Karate Network (KAN) [35]: This is a network of 34 members (nodes) of a Karate Club at a US university in the 1970s; there is an edge between two nodes if the corresponding members were seen interacting with each other during the observation period.

ISSN 2395-8618

(13) Dutch Literature 1976 Network (DLN) [36]: This is a network of 35 Dutch literary authors and critics (vertices) in 1976. There exists an edge between two vertices if one of them had made a judgment on the literature work of the author corresponding to the other vertex.

(14) Senator Press Release Network (SPN) [37]: This is a network of 92 US senators (vertices) during the period from 2007 to 2010. There exists an edge between two vertices if the corresponding senators had issued at least one joint press release.

(15) ModMath Network (MMN) [38]: This is a network of 38 school superintendents (vertices) in Allegheny County, Pennsylvania, USA during the 1950s and early 1960s. There exists an edge between two vertices if at least one of the two corresponding superintendents has indicated the other person as a friend in a research survey conducted to see which superintendents (who are in office for at least a year) are more influential to effectively spread around some modern Math methods among the school systems in the county.

(16) C. Elegans Neural Network (ENN) [39]: This is a network of 297 neurons (vertices) in the neural network of the hermaphrodite *Caenorhabditis Elegans*; there is an edge between two vertices if the corresponding neurons interact with each other (in the form of chemical synapses, gap junctions, and neuromuscular junctions).

(17) Word Adjacency Network (WAN) [40]: This is a network of 112 words (adjectives and nouns, represented as vertices) in the novel David Copperfield by Charles Dickens; there exists an edge between two vertices if the corresponding words appeared adjacent to each other at least once in the novel.

(18) Les Miserables Network (LMN) [41]: This is a network of 77 characters (nodes) in the novel Les Miserables; there exists an edge between two nodes if the corresponding characters appeared together in at least one of the chapters in the novel.

(19) Copperfield Network (CFN) [41]: This is a network of 87 characters in the novel David Copperfield by Charles Dickens; there exists an edge between two vertices if the corresponding characters appeared together in at least one scene in the novel.

(20) Graph and Digraph Glossary Network (GLN) [42]: This is a network of 72 terms (vertices) that appeared in the glossary prepared by Bill Cherowitzo on Graph and Digraph; there appeared an edge between two vertices if one of the two corresponding terms were used to explain the meaning of the other term.

(21) Centrality Literature Network (CLN) [43]: This is a network of 129 papers (vertices) published on the topic of centrality in complex networks from 1948 to 1979. There is an edge between two vertices if one of the two papers has cited the other paper as a reference.

(22) Citation Graph Drawing Network (GDN) [44]: This is a network of 311 papers (vertices) that were published in the Proceedings of the Graph Drawing (GD) conferences from 1994 to 2000 and cited in the papers published in the GD'2001 conference. There is an edge between two vertices if one of the two corresponding papers has cited the other paper as a reference.

(23) Anna Karenina Network (AKN) [41]: This a network of 138 characters (vertices) in the novel *Anna Karenina*; there exists an edge between two vertices if the corresponding characters have appeared together in at least one scene in the novel.

(24) Erdos Collaboration Network (ERN) [45]: This is a network of 472 authors (nodes) who have either directly published an article with Paul Erdos or through a chain of collaborators leading to Paul Erdos. There is an edge between two nodes if the corresponding authors have co-authored at least one publication.

We generate 100 instances of the configuration model-based random network graphs for each of the real-world network graphs. We compute the following seven node-level metrics on each of the real-world network graphs and the corresponding 100 instances of the random network graphs generated according to the Configuration model:

(i) Degree Centrality,
(ii) Eigenvector Centrality,
(iii) Closeness Centrality,
(iv) Betweenness Centrality,
(v) Clustering Coefficient,
(vi) Communicability and
(vii) Maximal Clique Size.

For each real-world network, we average the values for each of the above node-level metrics obtained for the 100 instances of the random network graphs with identical degree sequence. For each node-level metric, we then determine the Spearman's rank-based correlation coefficient between the values incurred for the metric in each of the 24 real-world network graphs and the average values for the metric computed based on the 100 instances of the corresponding configuration model-based random network graphs.

Table 2 lists the correlation coefficient values obtained for the seven node-level metrics for each real-world network graph and the corresponding configuration model-based instances of the random network graphs with an identical degree sequence. As expected, the correlation coefficient values for the degree centrality are either 0.99 or 1.0, vindicating identical degree sequence between the two graphs. The Communicability metric exhibits a very strong positive correlation for all the 24 network graphs. With respect to three

ISSN 2395-8618

TABLE II
CORRELATION COEFFICIENT VALUES FOR THE NODE-LEVEL METRICS BETWEEN THE REAL-WORLD NETWORK GRAPHS AND THE CORRESPONDING INSTANCES OF CONFIGURATION MODEL-BASED RANDOM NETWORK GRAPHS

#	Network	# nodes	$\lambda_{sp}^{RW}(k)$	Correlation Coefficient between Real-World Network Graphs and the Corresponding Configuration model-based Random Network Graphs						
				DegC	EVC	ClC	BWC	LCC	Comm.	MCS
1	MDN	62	1.04	1.00	0.99	0.99	0.91	0.11	0.99	0.58
2	CFN	58	1.11	1.00	0.99	0.99	0.79	0.29	0.99	0.93
3	HTN	113	1.21	1.00	0.99	0.99	0.90	0.54	0.99	0.85
4	FTC	48	1.21	1.00	0.79	0.82	0.77	0.31	0.82	0.38
5	SSC	24	1.22	0.99	0.67	0.68	0.86	0.24	0.83	0.22
6	PSN	238	1.22	1.00	0.98	0.95	0.83	0.11	0.98	0.42
7	MPN	35	1.23	1.00	0.86	0.83	0.90	0.08	0.87	0.46
8	RHF	217	1.27	1.00	0.87	0.88	0.89	0.00	0.87	0.36
9	UKF	81	1.35	1.00	0.93	0.90	0.86	0.10	0.93	0.67
10	WTM	80	1.38	0.99	0.98	0.98	0.97	0.55	0.98	0.72
11	JBN	198	1.45	1.00	0.90	0.90	0.72	0.30	0.90	0.75
12	KAN	34	1.47	0.99	0.88	0.73	0.87	0.10	0.89	0.61
13	DLN	35	1.49	1.00	0.89	0.87	0.71	0.66	0.90	0.56
14	SPN	92	1.57	1.00	0.96	0.95	0.86	0.31	0.96	0.72
15	MMN	38	1.59	0.99	0.81	1.00	0.82	0.75	0.87	0.80
16	ENN	297	1.68	1.00	0.86	0.72	0.95	0.63	0.86	0.62
17	WAN	112	1.73	1.00	0.93	0.84	0.95	0.67	0.93	0.66
18	LMN	77	1.82	1.00	0.84	0.67	0.84	0.37	0.85	0.81
19	CFN	87	1.83	0.99	0.95	0.64	0.97	0.40	0.95	0.81
20	GLN	72	2.01	1.00	0.79	0.61	0.92	0.65	0.81	0.64
21	CLN	129	2.03	1.00	0.96	0.98	0.91	0.62	0.96	0.86
22	GDN	311	2.24	1.00	0.79	0.88	0.79	0.81	0.81	0.73
23	AKN	138	2.48	1.00	0.95	0.72	0.94	0.33	0.95	0.85
24	ERN	472	3.00	1.00	0.89	0.92	0.78	0.71	0.89	0.75

centrality metrics (EVC, ClC, BWC), we observe a strong-very strong positive correlation for all the 24 network graphs, with the EVC exhibiting very strong positive correlation for 20 of the 24 real-world networks and the ClC and BWC metrics exhibiting very strong positive correlation for 17 and 18 of the 24 real-world networks respectively. The maximal clique size (MCS) metric exhibits strong-very strong positive correlation for 17 of the 24 real-world networks (very strongly positive correlation for 7 real-world networks and strongly positive correlation for 10 real-world networks). The local clustering coefficient (LCC) is the only node-level metric for which we observe a poor correlation between the real-world network graphs and the corresponding random network graphs with identical degree sequence. The level of correlation is very weak to at most moderate for 14 of the 24 real-world networks and very strongly positive for just one real-world network.

We summarize the above observations on the basis of the percentage chances of finding a real-world network graph with a very strong positive correlation with its corresponding configuration model-based random network graph (with an identical degree sequence) as follows: While there is a 100% chance (24 out of 24 networks) for a very strongly positive correlation in the case of communicability; for the three centrality metrics: the percentage chances of observing a very strongly positive correlation are respectively 83% (20 out of 24 networks) for EVC, 75% (18 out of 24 networks) for BWC and 71% (17 out of 24 networks) for ClC. In the case of maximal clique size and local clustering coefficient, the percentage chances of obtaining a very strong positive correlation are respectively 29% and 4%.

With respect to the impact of the spectral radius ratio for node degree on the correlation levels observed for the node-level metrics (see Figure 14), we observe the correlation levels for communicability and the centrality metrics to be independent of the spectral radius ratio for node degree. The correlation coefficient values for communicability and the centrality metrics (EVC, ClC and BWC) are consistently high (0.6 or above) for all the 24 real-world network graphs, irrespective of the values for the spectral radius ratio for node degree. In the case of both local clustering coefficient and maximal clique size, we observe the correlation levels to increase with increase in the spectral radius ratio for node degree (i.e., as the real-world networks are increasingly scale-free, we observe the correlation levels for these two metrics to increase with that of the configuration model-based random network graphs with identical degree sequence).

Table 3 lists the values for the three network-level metrics (spectral radius ratio for node degree, degree-based edge assortativity and algebraic connectivity) for the real-world network graphs and the corresponding configuration model-based random network graphs with identical degree sequence.

ISSN 2395-8618

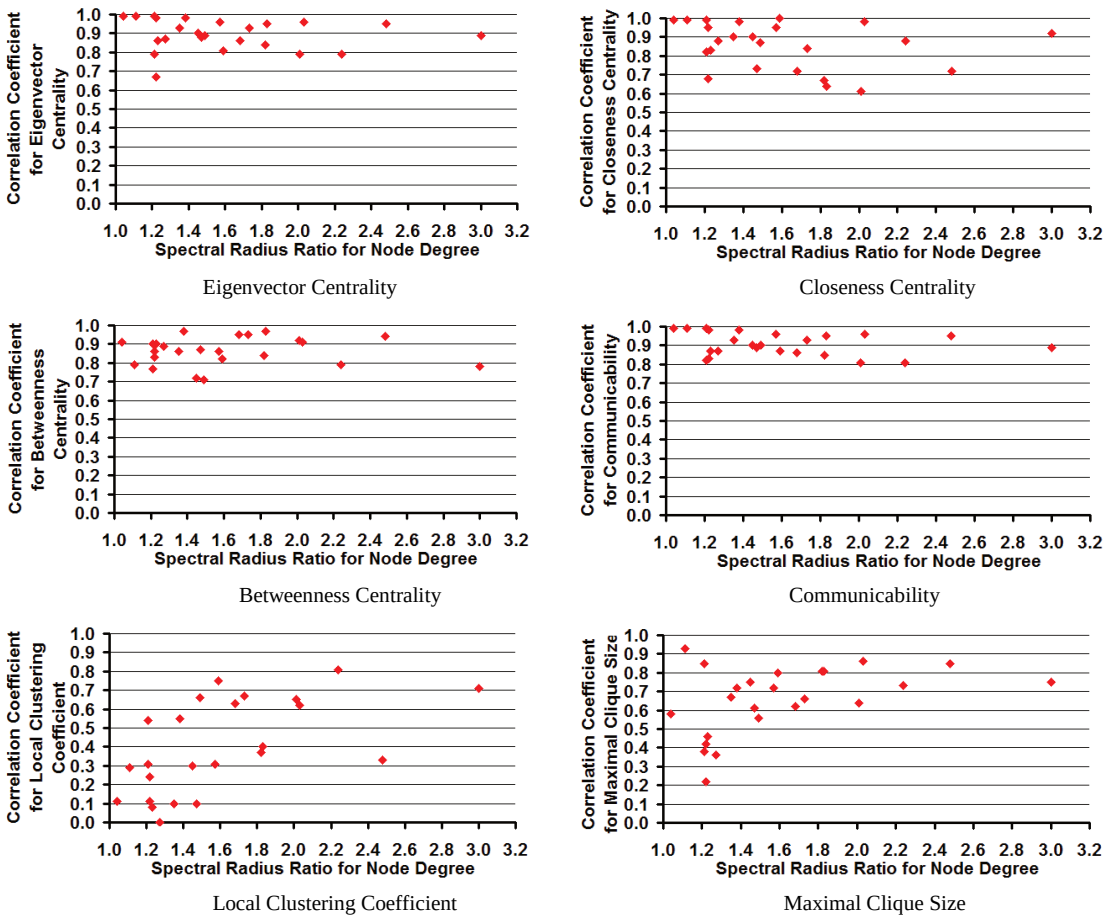


Fig. 14. Example to Illustrate the Computation of the Spearman's Rank-based Correlation Coefficient with respect to BWC on the Actual Graph and Configured Graph of Figure 3.

The value for a network-level metric reported for the random network graph corresponding to a real-world network graph is the average of the values for the metric evaluated on 100 instances of the random network graphs for the particular real-world network graph. In Table 3, we have colored the cells (in yellow) for which a graph (real-world network graph or the configuration model-based random network graph) incurs an equal or relatively larger value for the network-level metric. Figure 15 illustrates the distribution of the above three network-level metrics for both the real-world network graphs and the corresponding random network graphs. For each network-level metric, we evaluate the proximity of the data points to the diagonal line. The more closer are the data points to the diagonal line, the more closer are the values for the particular network-level metric for both the real-world network graph and the corresponding random network graph.

With respect to the spectral radius ratio for node degree, the general trend of the results is that for more than 20 of the 24 real-world network graphs, the spectral radius ratio for node degree is observed to be just marginally greater than the average value of the spectral radius ratio for node degree of the 100 corresponding instances of the configuration model-based random network graphs with identical degree sequence.

From Figure 14, except of a couple of real-world network graphs, we observe the data points for the spectral radius ratio for node degree to lie closer to the diagonal line, indicating both the real-world network graphs and their corresponding configuration model-based random network graphs incur comparable values for this metric, with the random network graphs consistently incurring slightly lower values for most cases.

With respect to the degree-based edge assortativity, we observe the configuration model-based random network graphs to incur negative values for the degree-based edge assortativity for all the 24 real-world network graphs analyzed (even though the edge assortativity values were positive for 1/3rd of the real-world network graphs). Thus, the configuration model-based random network graphs are highly likely to be dissortative even though their corresponding real-world network graphs are assortative. On the other hand, for real-world network graphs with negative values for the degree-based edge assortativity, we observe the edge assortativity of the corresponding configuration model-based random network

ISSN 2395-8618

TABLE III
COMPARISON OF THE VALUES FOR THE NETWORK-LEVEL METRICS: REAL-WORLD NETWORK GRAPHS AND THE CONFIGURATION MODEL-BASED RANDOM NETWORK GRAPHS WITH IDENTICAL DEGREE SEQUENCE

#	Network	Spectral Radius Ratio for Node Degree		Degree-based Edge Assortativity		Algebraic Connectivity	
		Real-World	Config-Random	Real-World	Config-Random	Real-World	Config-Random
1	MDN	1.04	1.03	–0.07	–0.05	1.67	1.66
2	CFN	1.11	1.10	–0.12	–0.07	0.58	0.58
3	HTN	1.21	1.20	–0.12	–0.10	1.00	0.99
4	FTC	1.21	1.18	–0.04	–0.05	0.68	0.87
5	SSC	1.22	1.20	–0.03	–0.12	0.15	0.44
6	PSN	1.22	1.18	0.22	–0.02	0.53	0.78
7	MPN	1.23	1.23	–0.17	–0.07	1.24	1.57
8	RHF	1.27	1.22	0.10	–0.02	1.71	1.88
9	UKF	1.35	1.30	0.00	–0.08	1.33	1.79
10	WTM	1.38	1.35	–0.39	–0.23	0.40	0.37
11	JBN	1.45	1.38	0.02	–0.06	0.57	0.93
12	KAN	1.47	1.55	–0.48	–0.18	0.47	0.56
13	DLN	1.49	1.42	0.07	–0.10	0.27	0.50
14	SPN	1.57	1.51	0.02	–0.08	0.64	0.84
15	MMN	1.59	1.51	0.04	–0.11	0.45	0.70
16	ENN	1.68	1.71	–0.16	–0.08	0.85	0.81
17	WAN	1.73	1.72	–0.13	–0.11	0.70	0.50
18	LMN	1.82	1.77	–0.16	–0.10	0.21	0.44
19	CFN	1.83	1.77	–0.26	–0.20	0.98	0.70
20	GLN	2.01	1.95	–0.16	–0.10	0.25	0.25
21	CLN	2.03	2.00	–0.20	–0.16	0.73	0.76
22	GDN	2.24	2.01	0.12	–0.02	0.12	0.29
23	AKN	2.48	2.42	–0.35	–0.28	0.33	0.42
24	ERN	3.00	2.50	0.18	–0.03	0.05	0.27

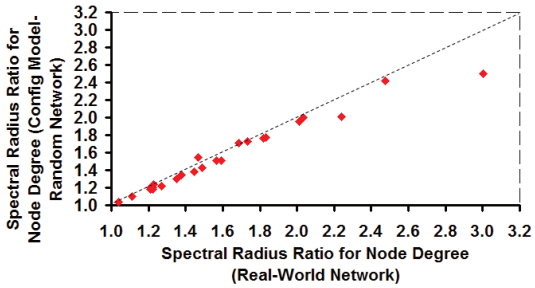
graphs to be relatively larger (i.e., farther away from -1). That is, for dissortative real-world network graphs, the corresponding configuration model-based random network graphs are likely to be relatively less dissortative. With respect to connectivity, we claim the configuration model-based random network graphs are more likely to exhibit relatively higher values for algebraic connectivity compared to their corresponding real-world network graphs (as is observed for 16 of the 24 real-world network graphs).

VI. RELATED WORK

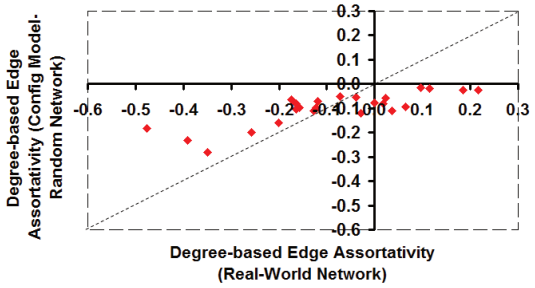
Degree preserving randomization [46] has been widely considered a technique of assessing whether the values for the node-level metrics and network-level metrics for a real-world network graph is just an artifact of the graph's inherent structural properties or properties that are unique for the nodes. Monte Carlo-based methods [47] were earlier used to generate random network graphs with identical degree sequence as that of the real-world networks. However, these methods were found to require a significant number of iterations (significantly larger than the number of edges in the real-world graphs) as well as are prone to introducing self-loops and multi-edges. The work in [48] formed the basis for modeling real-world social networks as random network graphs and evaluating the node-level metrics and network-level metrics for the two graphs. For certain social networks, network-level metrics such as the global clustering coefficient and the average path length were observed to be closer to that of the equivalent random networks and much different for others. The difference in the values for the network-level metrics is attributed to the sociological structure and influence among nodes not being captured in the random networks even though they are modeled to have an identical degree sequence to that of the social networks [48]. The authors in [49] conclude that global network-level metrics cannot be expected to be even closely reproduced in random network graphs generated with local constraints (such as the degree-preserving randomization). To corroborate this statement, degree-preserved randomized versions of the Internet at the level of ASs (AS - Autonomous Systems) are observed to have a fewer number of *k*-shells [50]. A *k*-shell is the largest sub graph of a graph such that the degree of every vertex is at least *k* within the sub graph [51]. The distribution of the *k*-shells has been observed to be dependent on the connectivity of the nodes in the network and hence the number of *k*-shells has been perceived to be a network-level metric [50].

Random networks generated from the traditional ER model have been observed to incur lower values for the local clustering coefficient (proportional to the probability of a link between any two nodes in the ER model) [5]; the correlation

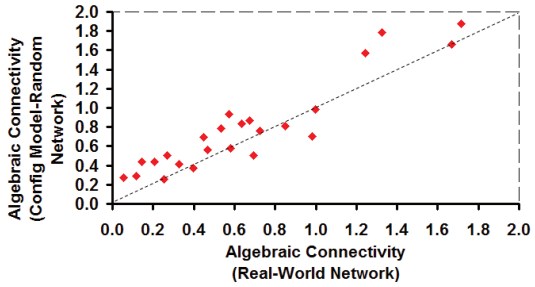
ISSN 2395-8618



Spectral Radius Ratio for Node Degree



Degree-based Edge Assortativity



Algebraic Connectivity

Fig. 15. Distribution of the Values for the Network-Level Metrics for Real-World Network Graphs and the Configuration Model-based Random Network Graphs with Identical Degree sequence.

coefficient analysis studies in this paper also indicate that the local clustering coefficient of the configuration model-based random network graphs do not correlate well with those of the corresponding real-world network graphs with an identical degree sequence. In [52], the authors proposed random graph models whose local clustering coefficient is as large as those observed for real-world networks as well as generate a power-law degree pattern [4] that could be controlled using certain operating parameters.

In another related study [53], degree-preserved random networks of a certain number of nodes and edges were observed to contain more feed forward loops (FFLs) when compared to the ER-random networks of the same number of nodes and edges; but the number of FFLs in the degree-preserved random networks has been observed to be significantly lower than the corresponding real-world biological networks. Degree preserving randomization was successfully applied in [54] to determine that clustering and modularity do not _ the number of driver nodes needed to effectively control real-world networks. In a related study, it was observed that the degree sequence of a real-world network was alone sufficient to generate a random network whose distribution for the control centrality [55] of a node was identical to that of the real-world network. The control centrality of a node [55] in a directed weighted network graph is a quantitative measure of the ability of a single node to control the entire network.

Even though several such studies have been conducted on degree-preserving randomization of real-world networks and analysis of the resulting random networks, no concrete information is available on the impact of the identical degree sequence on the centrality metrics (as well as the other node-level and network-level metrics considered in this paper such as communicability, maximal clique size, degree-based edge assortativity, algebraic connectivity and spectral radius ratio for node degree) for the real-world social networks and the equivalent degree-preserved random networks. We have to resort to a correlation-based study as the distribution profiles for a node-level metric in the real-world network graph and the corresponding degree-preserved random network graph is not sufficient to study the similarity in the ranking of the vertices in the two graphs with respect to the metric. To the best of our knowledge, ours is the first such study to comprehensively evaluate the similarity in the ranking of the vertices between the real-world network graphs and the corresponding degree-preserved random network graphs with respect to the centrality metrics and maximal clique size as well as to use the Spearman's rank-based correlation measure for correlation study in complex network analysis.

VII. Conclusions and Future Work

The results from Table 2 indicate that the ranking of the vertices in a real-world network graph with respect to the centrality metrics and communicability is more likely to be the same as the ranking of the vertices in a random network graph (generated according to the configuration model with an identical degree sequence) as a very strongly positive correlation (correlation coefficient values of 0.8 or above) is observed for a majority of the real-world networks analyzed. On the other hand, we observe that an identical degree sequence is not sufficient to increase the chances of obtaining an identical ranking of the vertices in a real-world network graph and its corresponding degree-preserved random network graph with respect to maximal clique size and local clustering coefficient. Thus, the maximal clique size and local clustering coefficient are node-level metrics that depend more on the network structure rather than on the degree sequence.

The results from Table 3 indicate that the spectral radius ratio for node degree is likely to be more for a real-world network graph vis-a-vis a random network graph with an identical degree sequence. On the other hand, we observe that all the degree-preserved random network graphs generated

ISSN 2395-8618

according to the configuration model are dissortative irrespective of the nature of assortativity of the corresponding real-world network graphs; nevertheless, the level of dissortativity is relatively less for degree-preserved random network graphs generated for real-world networks that are also dissortative. We observe that a random network graph is more likely to exhibit higher values for algebraic connectivity compared to the real-world network graph with which it has an identical degree sequence. As part of future work, we plan to run the community detection algorithms on the configuration model generated random network graphs and compare the modularity of the communities with those detected in the corresponding real-world network graphs with an identical degree sequence.

ACKNOWLEDGMENT

The research is financed by the NASA EPSCoR sub award NNX14AN38A from University of Mississippi.

REFERENCES

[1] P. Erdos and A. Renyi, "On Random Graphs I," *Publicationes Mathematicae*, vol. 6, pp. 290–297, 1959.

[2] M. D. Ugarte, A. F. Militino and A. T. Arnholt, *Probability and Statistics with R*, Chapman and Hall, 2nd Edition, August 2015.

[3] N. Meghanathan, "Spectral Radius as a Measure of Variation in Node Degree for Complex Network Graphs," *Proceedings of the 3rd International Conference on Digital Contents and Applications*, (DCA 2014), pp. 30–33, Hainan, China, December 20–23, 2014.

[4] A-L. Barabasi and R. Albert, "Emergence of Scaling in Random Networks," *Science*, vol. 286, no. 5439, pp. 509–512, October 1999.

[5] M. Newman, *Networks*: *An Introduction*, Oxford University Press, 1st edition, May 2010.

[6] T. Britton, M. Deijfen and A. Martin-Lof, "Generating Simple Random Graphs with Prescribed Degree Distribution," *Journal of Statistical Physics*, vol. 124, no. 6, pp. 1377–1397, September 2006.

[7] P. Bonacich, "Power and Centrality: A Family of Measures," *American Journal of Sociology*, vol. 92, no. 5, pp. 1170–1182, February 1987.

[8] L. Freeman, "A Set of Measures of Centrality based on Betweenness," *Sociometry*, vol. 40, no. 1, pp. 35–41, February 1977.

[9] L. Freeman, "Centrality in Social Networks Conceptual Clarification," *Social Networks*, vol. 1, no. 3, pp. 215–239, 1979.

[10] M. E. J. Newman, "The Structure and Function of Complex Networks," *SIAM Review*, vol. 45, no. 2, pp. 167–256, 2003.

[11] E. Estrada, "Community Detection based on Network Communicability," *Chaos*, vol. 21, no. 1, p016103, March 2011.

[12] N. Meghanathan, "Distribution of Maximal Clique Size of the Vertices for Theoretical Small-World Networks and Real-World Networks," *International Journal of Computer Networks and Communications*, vol. 7, no. 4, pp. 21–41, July 2015.

[13] M. E. J. Newman, "Assortative Mixing in Networks," *Physical Review Letters*, vol. 89, no. 2, 208701, November 2002.

[14] M. F. Triola, *Elementary Statistics*, 12th Edition, Pearson, NY, USA, December 2012.

[15] D. C. Lay, S. R. Lay and J. J. McDonald, *Linear Algebra and its Applications*, 5th edition, Pearson Publishers, January 2015.

[16] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, *Introduction to Algorithms*, 3rd edition, MIT Press, July 2009.

[17] JAMA (2016, January 29). http://math.nist.gov/javanumerics/jama/.

[18] B. Pattabiraman, M. A. Patwary, A. H. Gebremedhin, W. K. Liao and A. Choudhary, "Fast Algorithms for the Maximum Clique Problem on Massive Sparse Graphs," *Proceedings of the 10th International Workshop on Algorithms and Models for the Web Graph*: *Lecture Notes in Computer Science*, vol. 8305, pp. 156–169, Cambridge, MA, USA, December 2013.

[19] B. G. Horne, "Lower Bounds for the Spectral Radius of a Matrix," *Linear Algebra and its Applications*, vol. 263, pp. 261–273, September 1997.

[20] F. R. K. Chung, *Spectral Graph Theory*, American Mathematical Society, 1st edition, December 1996.

[21] N. M. M. Abreu, "Old and New Results on Algebraic Connectivity of Graphs," *Linear Algebra and its Applications*, vol. 423, no. 1, pp. 53–73, May 2007.

[22] M. Fiedler, "Algebraic Connectivity of Graphs," *Czechoslovak Mathematical Journal*, vol. 23, no. 98, pp. 298–305, 1973.

[22] M. Fiedler, "Algebraic Connectivity of Graphs," *Czechoslovak Mathematical Journal*, vol. 23, no. 98, pp. 298–305, 1973.

[23] J. D. Evans, *Straightforward Statistics for the Behavioral Sciences*, 1st Edition, Brooks Cole Publishing Company, August 1995.

[24] Y. Takahata, "Diachronic Changes in the Dominance Relations of Adult Female Japanese Monkeys of the Arashiyama B Group," *The Monkeys of Arashiyama*, pp. 124–139, Albany: State University of New York Press, 1991.

[25] H. R. Bernard, P. D. Killworth and L. Sailer, "Informant Accuracy in Social Network Data IV: A Comparison of Clique-level Structure in Behavioral and Cognitive Network Data," *Social Networks*, vol. 2, no. 3, pp. 191–218, 1980.

[26] L. Isella, J. Stehle, A. Barrat, C. Cattuto, J. F. Pinton and W. Van den Broeck, "What's in a Crowd? Analysis of Face-to-Face Behavioral Networks," *Journal of Theoretical Biology*, vol. 271, no. 1, pp. 166–180, February 2011.

[27] J. L. Moreno, *The Sociometry* Reader, pp. 534–547, The Free Press, Glencoe, IL, USA, 1960.

[28] J. H. Michael, "Labor Dispute Reconciliation in a Forest Products Manufacturing Facility," *Forest Products Journal*, vol. 47, no. 11–12, pp. 41–45, October 1997.

[29] V. Gemmetto, A. Barrat and C. Cattuto, "Mitigation of Infectious Disease at School: Targeted Class Closure vs. School Closure," *BMC Infectious Diseases*, vol. 14, no. 695, pp. 1–10, December 2014.

[30] J. Gil-Mendieta and S. Schmidt, "The Political Network in Mexico," *Social Networks*, vol. 18, no. 4, pp. 355–381, October 1996.

[31] L. C. Freeman, C. M. Webster and D. M. Kirke, "Exploring Social Structure using Dynamic Three-Dimensional Color Images," *Social Networks*, vol. 20, no. 2, pp. 109–118, April 1998.

[32] T. Nepusz, A. Petroczi, L. Negyessy and F. Bazso, "Fuzzy Communities and the Concept of Bridgeness in Complex Networks," *Physical Review E*, vol. 77, no. 1, 016107, January 2008.

[33] D. A. Smith and D. R. White, "Structure and Dynamics of the Global Economy: Network Analysis of International Trade 1965–1980," *Social Forces*, vol. 70, no. 4, pp. 857–893, June 1992.

[34] P. Geiser and L. Danon, "Community Structure in Jazz," *Advances in Complex* Systems, vo. 6, no. 4, pp. 563–573, July 2003.

[35] W. W. Zachary, "An Information Flow Model for Conflict and Fission in Small Groups," *Journal of Anthropological Research*, vol. 33, no. 4, pp. 452–473, 1977.

[36] W. de Nooy, "A Literary Playground: Literary Criticism and Balance Theory," *Poetics*, vol. 26, no. 5–6, pp. 385–404, August 1999.

[37] J. Grimmer, "A Bayesian Hierarchical Topic Mode for Political Texts: Measuring Expressed Agendas in Senate Press Releases," *Political Analysis*, vol. 18, no. 1, pp. 1–35, January 2010.

[38] R. O. Carlson, *Adoption of Educational Innovations*, Center for Advanced Study of Educational Admin, 1971.

[39] J. G. White, E. Southgate, J. N. Thomson and S. Brenner, "The Structure of the Nervous System of the Nematode *Caenorhabditis Elegans*," *Philosophical Transactions B*, vol. 314, no. 1165, pp. 1–340, November 1986.

ISSN 2395-8618

[40] M. E. J. Newman, "Finding Community Structure in Networks using the Eigenvectors of Matrices," *Physical Review E*, vol. 74, no. 3, 036104, September 2006.

[41] D. E. Knuth, *The Stanford GraphBase*: *A Platform for Combinatorial Computing*, 1st Edition, Addison-Wesley, Reading, MA, December 1993.

[42] UFL Datasets (2016, January 29). http://www.cise.ufl.edu/research/sparse/matrices/Pajek/GlossGT.html.

[43] N. P. Hummon, P. Doreian and L. C. Freeman, "Analyzing the Structure of the Centrality-Productivity Literature Created between 1948 and 1979," *Science Communication*, vol. 11, no. 4, pp. 459–480, May 1990.

[44] T. Biedl and B. J. Franz, "Graph-Drawing Contest Report," *Proceedings of the 9th International Symposium on Graph Drawing*, pp. 513–521, September 2001.

[45] Pajek (2016, January 29). http://vlado.fmf.uni-lj.si/pub/networks/data/.

[46] A. R. Rao, R. Jana and S. Bandyopadhyay, "A Markov Chain Monte Carlo Method for Generating Random (0, 1)-Matrices with given Signals," *Sankhya*: *The Indian Journal of Statistics*, *Series A*, vol. 58, no. 2, pp. 225–242, June 1996.

[47] F. Viger and M. Latapy, "Efficient and Simple Generation of Random Simple Connected Graphs with Prescribed Degree Sequence," *Proceedings of the 11th Annual International Conference on Computing and Combinatorics*, pp. 440–449, Kunming, China, August 2005.

[48] M. E. Newman, D. J. Watts and S. H. Strogatz, "Random Graph Models of Social Networks," *Journal of the National Academy of Sciences USA*, vol. 99, pp. 2566–2572, February 2002.

[49] C. Orsini, M. M. Dankulov, P. Colomer-de-Simon, A. Jamakovic, P. Mahadevan, A. Vahdat, K. E. Bassler, Z. Toroczkai, M. Boguna, G. Caldarelli, S. Fortunato and D. Krioukov, "Quantifying Randomness in Real Networks," *Nature Communications*, vol. 6, no. 8627, pp. 1–10, October 2015.

[50] M. Kitsak, L. K. Gallos, S. Havlin, F. Liljeros, L. Muchnik, H. E. Stanley and H. A. Makse, "Identification of Influential Spreaders in Complex Networks," *Nature Physics*, vol. 6, pp. 888–893, August 2010.

[51] D. Miorandi and F. De Pellegrini, "K-Shell Decomposition for Dynamic Complex Networks," *Proceedings of the 8th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, pp. 488–496, Avignon, France, June 2010.

[52] R. Elsasser and A. Neubert, "Toward Proper Random Graph Models for Real World Networks," *Proceedings of the 9th International Conference on Networks*, pp. 306–315, Menuires, France, April 2010.

[53] S. Itzkovitz, R. Milo, N. Kashtan, G. Ziv and U. Alon, "Subgraphs in Random Networks," *Physical Review E*, vol. 68, no. 2, 026127, August 2003.

[54] M. Posfai, Y-Y. Liu, J-J. Slotine and A-L. Barabasi, "Effect of Correlations on Network Controllability," *Scientific Reports*, vol. 3, no. 1067, pp. 1–21, January 2013.

[55] Y-Y. Liu, J-J. Slotine and A-L. Barabasi, "Control Centrality and Hierarchical Structure in Complex Networks," *PLoS One*, vol. 7, no. 9, e44459, September 2012.

ISSN 2395-8618

Ajuste de filtros wavelets usando inteligencia artificial para compresión de imágenes

Ignacio Hernández Bautista, Jesús Ariel Carrasco-Ochoa, José Francisco Martínez-Trinidad,
Oscar Camacho-Nieto, Oleksiy Pogrebnyak

***Resumen*—Se presenta un método para compresión de imágenes sin pérdidas, usando la transformada wavelet lifting con ajuste automático de coeficientes de filtros wavelets para una mayor compresión sin pérdidas. La propuesta se basa en reconocimiento de patrones utilizando clasificador 1-NN. Utilizando el reconocimiento de patrones se optimizan los coeficientes de los filtros lifting de manera global para cada imagen. La técnica propuesta fue aplicada para la compresión de imágenes de prueba y comparada con los filtros wavelets estándares CDF (2,2) y CDF (4,4), obteniendo resultados mejores en relación a la entropía obtenida para cada imágenes, así, como para el promedio general.**

***Palabras clave*—Compresión de imágenes sin pérdida, esquema lifting, wavelets, reconocimiento de patrones.**

Adjustment of Wavelet Filters for Image Compression Using Artificial Intelligence

***Abstract*—A method for lossless image compression using wavelet lifting transform with automatic adjustment of wavelet filter coefficients for better compression is presented. The proposal is based on pattern recognition by 1-NN classifier. Using the pattern recognition, the lifting filter coefficients are optimized globally for each image. The proposed technique was aplied to test images and the compression results were compared to the results produced by the standard CDF (2,2) and CDF (4,4) wavelet filters. The results obtained with the optimized wavelet filters are better in terms of the achieved entropy for all images, as well as for the overall performance.**

***Index Terms*—Lossless image compression, wavelets, lifting scheme, pattern recognition.**

Manuscrito recibido el 02 de febrero de 2016, aceptado para la publicación el 27 de mayo de 2016, publicado el 25 de junio de 2016.

Ignacio Hernández Bautista, Jesús Ariel Carrasco-Ochoa, José Francisco Martínez-Trinidad están con el Departamento de Ciencias Computacionales, Instituto Nacional de Astrofísica, Óptica y Electrónica, Luis Enrique Erro #1, Santa María Tonantzintla, Puebla, México (correo: ignaciohb@gmail.com, ariel@inaoep.mx, fmartine@inaoep.mx).

Oscar Camacho-Nieto está con el Instituto Politécnico Nacional, CIDETEC-IPN, Av. Juan de Dios Bátiz s/n, DF, México (correo: oscarc@cic.ipn.mx).

Oleksiy Pogrebnyak está con el Instituto Politécnico Nacional, CIC-IPN, Av. Juan de Dios Bátiz s/n, DF, México (correo: olek@cic.ipn.mx).

I. Introducción

En las pasadas dos décadas, la transformada wavelets se convirtió en una herramienta popular y poderosa para diferentes aplicaciones de procesamiento señales/imágenes como cancelación del ruido, detección de rasgos, compresión de datos, etc. No obstante, todavía se presenta el problema de selección adecuada o diseño de wavelet para un problema dado. A diferencia de muchas transformadas ortogonales, la transformada wavelets puede usar diferentes funciones de base, y es importante que la base seleccionada dé la mejor representación de la señal analizada. Por ejemplo, en aplicaciones de detección de señales donde la transformada wavelets calcula la correlación cruzada entre la señal y wavelet, la wavelet acomodada para la señal resulta en un pico grande en el dominio de la transformada. En las aplicaciones de compresión de señales, el acoplamiento de wavelet y señal resulta en una mejor representación de la señal, y por ello, en el nivel de compresión más alto.

La técnica estándar para encontrar bases ortonormales de wavelets con soporte compacto es de Daubechies [1], y en el caso de bases biortogonales la solución clásica para wavelets de soporte compacto y regularidad arbitraria fue dada por Cohen, Daubechies y Feauveau en [2]. Desafortunadamente, estas técnicas no son prácticas por su complejidad e independencia de la señal analizada.

A. Tewfik *et al*. en [3] propuso el primer método para encontrar las bases wavelets ortonormales óptimas para señales de voz usando su parametrización. Este método se limita a un número finito de escalas y la optimización se realiza en el dominio de tiempo, de una manera recurrida.

Otras técnicas existentes no tratan de diseñar la wavelet directamente, algunas de ellas a diferencia usan un banco de las wavelets diseñadas previamente [4], algunos métodos acoplan las wavelets a la señal proyectándola a las wavelets existentes o transformando las bases de wavelets [5].

Entre técnicas recientes de acoplamiento de wavelets más interesantes se encuentran propuestas por Chapa y Rao [6] A. Gupta *et al.* [7]. En [6], los autores adaptan la wavelet generalizada de Meyer, minimizando la diferencia entre la wavelet y espectro de la señal, pero su método es complejo computacionalmente porque requiere una sincronización de fase, y es diseñado sólo para la detección de señales determinísticas. En [7], el método para estimación de filtros pasa altas de análisis wavelets se propone para señales que

ISSN 2395-8618

representan procesos similares de sí mismos y cuya autocorrelación se puede modelar analíticamente usando el índice de similitud propia. Aunque los autores afirman que su método es simple, se requiere un estimador de máxima verosimilitud del índice de similitud propia; además. Este método generalmente produce bancos de filtros que no tienen la propiedad de restauración perfecta de la señal.

En este artículo, se considera la aplicación de la transformada discreta wavelets para la compresión de imágenes sin pérdidas de información. Para tal aplicación, ampliamente se usa la transformada rápida wavelets que se implementa con esquema lifting propuesta por W. Sweldens [8], como un método estándar para la transformada de números enteros [9]. Los usados en el esquema lifting filtros wavelets se obtienen de los filtros de wavelets biortogonales conocidos factorizando las matrices polifase [8, 10]. Para la compresión sin pérdidas de imagines digitales, los más populares son filtros de lifting de un solo paso CDF (2,2) y CDF (4,4) [9].

Se conocen varias pruebas de mejorar el rendimiento de la compresión de imágenes sin pérdidas ajustando parámetros de filtros lifting, ya que estos filtros son fáciles de modificar obteniendo así la restauración perfecta de la imagen original [11–15]. Como regla, los autores de estos trabajos tratan de optimizar el filtro predictor lifting minimizando el error cuadrático promedio de predicción en la salida, minimizando así la energía de los coeficientes en el dominio de la transformada. Con ello, autores [11, 12, 14] obtienen (a veces, dependiendo de los datos) unos resultados positivos combinando con otras mejoras, pero H. Thielemann [13] reportó resultados negativos de la optimización de mínimos cuadrados en comparación con el rendimiento de filtros lifting CDF (2,2). Es interesante destacar, que la técnica propuesta en [14] optimiza el predictor lifting con el mismo criterio del error cuadrático promedio de predicción pero haciendo uso de diferencias de señal para calcular la autocorrelación sobre estas diferencias y no sobre la señal misma que es la técnica de cálculo estándar.

Una alternativa a la optimización minimizando el error cuadrático promedio fue propuesta en [15], donde la minimización del error de predicción se realiza usando norma ℓ_1 en lugar de norma ℓ_2, en otras palabras, minimizando el valor absoluto del error en lugar de energía del error. Además, los autores de este artículo propusieron minimizar la norma ℓ_2 de la diferencia entre aproximaciones de señal en la salida del filtro lifting de renovación y la salida del filtro ideal pasa bajas. Desafortunadamente, la minimización de norma ℓ_1 es mucho más compleja que la minimización de norma ℓ_2 y requiere técnicas sofisticadas. Los autores consideraron el caso de compresión de imagines sin perdidas, y los resultados obtenidos muestran comportamiento un poco mejor comparando con los filtros lifting CDF (2,2). Por otra parte, la complejidad del algoritmo lifting adaptativo es muy alta.

En el presente artículo, nosotros proponemos usar técnicas de inteligencia artificial para mejorar el rendimiento de los filtros wavelets lifting; se propone desarrollar un modelo para el diseño automático de filtros wavelet para la compresión de imágenes sin pérdidas. Por lo tanto, en este artículo nosotros consideramos la aplicación de los algoritmos de clasificación de patrones como *1-NN*, para clasificar de una manera global los coeficientes espectrales de cada imagen a comprimir en el dominio de la transformada discreta de coseno. Como resultado; se obtienen los coeficientes para filtros wavelet lifting de predicción y de renovación que realizan la transformada wavelets de las imágenes obteniéndose la tasa de compresión sin pérdidas, mayor que producen los filtros wavelets lifting estándares CDF (2,2) y CDF (4,4).

Este artículo está organizado de la siguiente forma: En la sección 2 se describe la generalización del esquema lifting, de la cual se formulan las condiciones necesarias para las modificaciones de los coeficientes de filtros wavelet lifting. En la sección 3 se describe la generalización del método de reconocimiento de patrones *K-NN.* En la sección 4 se presenta el método propuesto con el cual se obtienen los coeficientes del esquema lifting para la transformada wavelet discreta. En la sección 5 se muestran los resultados obtenidos para el conjunto inicial de las diferentes imágenes iniciales de prueba. Para terminar con las conclusiones en la sección 6.

II. GENERALIZACIÓN DEL ESQUEMA LIFTING

El algoritmo clásico de la transformada discreta wavelets (DWT) de banco de filtros de Vetterli-Mallat [16] [17] [18] para la descomposición y reconstrucción de una señal de 1 etapa puede ser descrito como sigue: la señal de entrada de una dimensión (1D) es filtrada y sobremuestreada siendo descompuesta en dos partes: una señal pasa-baja (LP) y una señal pasa-alta (HP). En el caso de la descomposición de una imagen, los pasos de procesamiento usando un algoritmo compuesto de un banco de filtros se representa por el esquema de descomposición subbanda y se describen como:

a) aplicar los filtros LP y HP, submuestreando sobre las filas;
b) la señal que fue filtrada sobre las filas, es pasada a través de dos filtros de 1D, un LP y un HP y submuestreada sobre las columnas.

Con ello, se obtienen los 4 cuadrantes de los coeficientes de la imagen transformada: 3 cuadrantes de los detalles horizontales, verticales y diagonales. Y el cuadrante que corresponde a la señal pasada por dos filtros LP, de filas y columnas, son aproximaciones y sirve como entrada para otro nivel de descomposición; tal procedimiento se repite hasta que el tamaño de aproximaciones alcanza el tamaño de los filtros LP o HP.

En el caso de la transformada rápida wavelets, o transformada wavelets lifting, la descomposición de una señal discreta de 1D $\mathbf{s} = \{s_k\}, k = 1,\ldots, N$ y la wavelet más sencilla, la wavelet de Haar puede ser descrita como sigue [8] [19]:

ISSN 2395-8618

La primera etapa consiste en dividir la señal **S** en muestras pares e impares: $\{d_j\}$ y $\{e_j\}$. Durante la segunda etapa, la predicción, las muestras impares son predichas usando interpolación lineal, como sigue:

$$d_j = d_j - e_j,\ j = 1,..., N/2 \tag{1}$$

Durante la tercera etapa las muestras pares son actualizadas también con el fin de preservar el valor promedio de las muestras. Para esto se usa la siguiente expresión:

$$e_j = e_j + \frac{1}{2} d_j,\ j = 1,..., N/2 \tag{2}$$

Los próximos niveles de descomposición de la DWT son obtenidos aplicando el esquema de lifting a los datos actualizados $\{e_k\}$ de la señal original.

La transformada inversa para el esquema lifting para nuestro ejemplo es como sigue:

– actualización de datos inversa:

$$d_j = d_j + e_j,\ j = 1,..., N/2 \tag{3}$$

– predicción inversa:

$$d_j = d_j + e_j,\ j = 1,..., N/2 \tag{4}$$

– composición de la señal de salida:

$$\begin{aligned} x_{2i} &= e_j,\ i = 2,..., N,\ j = 1,..., N/2 \\ x_{2i+1} &= d_j,\ i = 1,..., N,\ j = 1,..., N/2 \end{aligned} \tag{5}$$

El ejemplo considerado presenta el esquema lifting de un nivel de descomposición y reconstrucción de la señal para la bien conocida wavelet Haar, también conocida como la wavelet de Cohen-Daubechies-Feauveau de primer orden con un momento de desvanecimiento, o CDF (1,1) [2]. La forma wavelet más popular para la compresión de imágenes sin pérdidas de información de este tipo es la CDF (2,2), la cual posee dos momentos desvanecidos para ambas la wavelet primitiva y la wavelet dual. Las etapas de predicción y actualización de los datos del esquema lifting para el caso de wavelet CDF (2,2) de números enteros son las siguientes [9]:

$$d_j = d_j - \left\lfloor \frac{1}{2}\left(e_j + e_{j-1}\right) \right\rfloor,\ j = 1,..., N/2 \tag{6}$$

$$e_j = e_j + \left\lfloor \frac{1}{4}\left(d_j + d_{j-1}\right) \right\rfloor,\ j = 1,..., N/2 \tag{7}$$

donde $\lfloor \cdot \rfloor$ denota la operación del redondeo al número entero más cercano [9]. Para la síntesis de la señal original, las sumas se cambian por restas, y restas por sumas.

Como fue mencionado anteriormente, la ventaja principal del esquema lifting es que se requieren 2 veces menos operaciones que el algoritmo clásico de banco de filtros. Además, el esquema lifting permite redondear los valores en las etapas de predicción y actualización de datos. Esto permite operar con números enteros, lo cual es importante para las aplicaciones de compresión de datos sin pérdidas de información.

Las funciones de transferencia correspondientes de filtros lifting de orden (4,4) se pueden escribir, como:

$$H_p(z) = 1 + p_0\left(z + z^{-1}\right) + p_1\left(z^2 + z^{-2}\right) \tag{8}$$

$$H_u(z) = 1 + H_p(z) u_0\left(z + z^{-1}\right) + H_p(z) u_1\left(z^3 + z^{-3}\right) \tag{9}$$

El filtro de la ecuación (8) es el filtro de predicción lifting, entonces, es un filtro pasa altas que tiene que cumplir con la condición de tener un cero el punto $z = 1$ del plano complejo Z que corresponde a la frecuencia $\omega = 0$ [20]. Esto significa, que la condición de admisibilidad los filtros wavelet lifting de predicción es:

$$p_0 + p_1 = -\frac{1}{2} \tag{10}$$

Con los valores de $\{p_i\}$ que cumplen la condición $H_p(z)\big|_{z=1} = 0$ dados por la ecuación (10), podemos encontrar

$$H_p(-1) = 2 \tag{11}$$

$$H_p(0) = 1 \tag{12}$$

Lo que significa, que el filtro de predicción tiene ganancia 2 en la frecuencia más alta $\omega = \pi$ y una ganancia unitaria en la frecuencia $\omega = \frac{\pi}{2}$, la cual corresponde a la mitad de la banda de transición del filtro. Entonces, si se pide la respuesta en frecuencia del filtro normalizada, el factor de normalización debe ser $\frac{1}{2}$.

Siguiendo la técnica del análisis considerada, los coeficientes del filtro de la actualización, que es un filtro paso bajas, se pueden encontrar. De esta manera, el filtro paso bajas $H_u(z)$ debe tener un cero en el punto $z = -1$ del plano complejo **Z** que corresponde a la frecuencia $\omega = \pi$ [20]. Esta condición se cumple cuando:

$$u_0 + u_1 = \frac{1}{4} \tag{13}$$

La ecuación anterior (13) es la condición de admisibilidad para los filtros escala lifting (filtros de actualización, o renovación). Cuando los coeficientes del filtro $H_u(z)$ satisfacen la condición (13), entonces, $H_u(1)|_{z=1}=1$ lo que significa que el filtro tiene ganancia unitaria en la frecuencia $\omega = 0$.

Una simplificación elegante de fórmulas (10) y (13), para el caso de lifting de orden (4,4) se propuso en [21]. Las fórmulas

ISSN 2395-8618

para los coeficientes de filtros wavelet se muestran a continuación:

$$p_0 = -\frac{128-a}{256}, \quad p_1 = \frac{a}{256} \tag{14}$$

$$u_0 = \frac{64-b}{256}, \quad u_1 = -\frac{b}{256} \tag{15}$$

Donde a y b son los parámetros que controlan las propiedades de la transformada wavelet. También, en [21] se encontró la correspondencia entre estos parámetros de control y los filtros wavelet convencionales (que no usan lifting). Con esta correspondencia, los filtros estándar CDF(2,2) tienen coeficientes con valores $a=0$ y $b=0$, y los coeficientes de los filtros CDF(4,4) son $a=16$ y $b=8$. Tal caracterización de cada filtro wavelet por un solo coeficiente entero permita guardar estos coeficientes junto con los datos de imagen sin incrementar mucho el tamaño del archivo resultante.

III. Clasificador *k*-NN

El método de *k* vecinos más cercanos (*k*-NN) [22] es un método de clasificación supervisada [23]. El método *K-NN* es uno de los algoritmos de clasificación más eficientes y a la vez más simples que existen; este algoritmo está basado en el enfoque de métricas y está fundado en la suposición de que los patrones cercanos entre si pertenecen a la misma clase y por ello un nuevo patrón a clasificar se determina la proximidad de este patrón por medio de alguna medida de similitud. Generalmente, se usa la distancia Euclidiana, y se va calculando la distancia con respecto a los n patrones ya existentes dentro del conjunto fundamental. Se clasifica a la clase del patrón *k* más cercano, donde *k* es un entero positivo generalmente impar. Desde los artículos pioneros de este método hasta modificaciones al método original como los presentados en [24], [25], [26], [27], [28] en los cuales se presentan variación del método original o la combinación de clasificadores. Los cuales han demostrado en las diferentes aplicaciones que se ha utilizado este método, que es uno de más eficientes que existen para el reconocimiento y clasificación de patrones.

El algoritmo a seguir para $k=1$ es el siguiente:

1. Se escoge una métrica a utilizar (normalmente la Euclidiana).
2. Se calculan las distancias de un patrón **x** desconocido por clasificar, a cada uno de los patrones del conjunto fundamental.
3. Se obtiene la distancia mínima.
4. Se asigna al patrón **x** la clase del patrón con la mínima distancia.

Cuando se usa un k mayor a 1, se sigue el mismo procedimiento antes descrito, tan solo que para asignar la clase del patrón a clasificar se usa la regla de mayoreo.

IV. Modelo propuesto de la compresión de imágenes

En esta sección se describirá de forma general el método propuesto y los pasos para la obtención automática de los coeficientes de los filtros wavelets.

El método propuesto se puede describir por los pasos del algoritmo:

1. Primer paso del algoritmo propuesto es calcular el espectro de potencia de la imagen adquirida.
2. En el siguiente paso, el espectro se analiza usando técnicas de inteligencia artificial, para obtener los coeficientes de los filtros wavelets lifting.
3. Teniendo ya los coeficientes de los filtros wavelets se aplica la transformada discreta wavelet lifting, la cual permita reducir la entropía de los datos.
4. Finalmente, la imagen transformada se procesa con una de las técnicas existentes de búsqueda de los árboles de los coeficientes wavelets diferentes de cero [29] y se codifica con uno de los codificadores de entropía existentes [30].

En la etapa 4 se realiza el propio proceso de compresión de la imagen, pero en el presente artículo nosotros concentraremos más en los pasos 1) – 3).

El primer paso del algoritmo propuesto es aplicar a la imagen analizada **s** de tamaño $M \times N$ la transformada discreta del coseno (DCT) [20], para obtener el espectro de potencia promediado, como:

$$S(i) = \frac{\alpha(i)}{MN}\left[\sum_{l=0}^{N-1}\sum_{q=0}^{M-1} s(l,q)\cos\left(\pi\frac{i(2q+1)}{2M}\right)\right]^2 \quad para \quad 0 \le i \le M-1 \tag{16}$$

donde: *M* es el número de filas y *N* es número de columnas que contenga la imagen a procesar,

$$\alpha(i) = \begin{cases} 1, & 1 \le i \le M-1 \\ \dfrac{1}{\sqrt{2}}, & i = 0 \end{cases}$$

Posteriormente se obtiene el vector resultante x_i, interpolado para tener longitud fija de 16 elementos:

$$\mathbf{x} = F_{16}^{-1}\left\{F_M\left\{S(i)\right\}\right\} \tag{17}$$

donde $F_M\{\cdot\}$ denota la transformada directa de Fourier de tamaño *M,* $F_{16}^{-1}\{\cdot\}$ es la transformada inversa de Fourier de tamaño 16.

Con ello se obtiene el vector característico y reducido **x** de la imagen. Con este vector se va generando el método de aprendizaje y clasificación utilizando el clasificador *1-NN*, en conjunto con los coeficientes de los filtros wavelets previamente seleccionados por la búsqueda exhaustiva.

Los coeficientes utilizados para la generación de los patrones para entrenamiento de clasificación y recuperación

ISSN 2395-8618

son los coeficientes de filtros wavelets lifting de predicción y renovación (8), (9), obtenidos variando coeficientes a, b de la generalización de filtros usada (14), (15). Los coeficientes a, b para formar el conjunto de patrones de entrenamiento fueron obtenidos con la búsqueda exhaustiva de los coeficientes a, b óptimos con el criterio de mínima entropía de tren de bits de datos en el dominio de transformada wavelet; está métrica fue propuesta en [9] para comparar diferentes algoritmos de compresión sin pérdidas:

$$H_q(\tilde{\mathbf{s}}) = -\frac{H(\tilde{\mathbf{s}}_a) + \sum_{q=1}^{Q}\left[H(\tilde{\mathbf{s}}_h) + H(\tilde{\mathbf{s}}_v) + H(\tilde{\mathbf{s}}_d)\right]}{MN} \tag{18}$$

donde $H(\tilde{\mathbf{s}}_a)$ es la entropía de Shannon de las aproximaciones $\tilde{\mathbf{s}}_a$, $H(\tilde{\mathbf{s}}_h), H(\tilde{\mathbf{s}}_v), H(\tilde{\mathbf{s}}_d)$ son entropías de los coeficientes de cada cuadrante de q-ésimo nivel de descomposición wavelets (detalles horizontales $\tilde{\mathbf{s}}_h$, verticales $\tilde{\mathbf{s}}_v$ y diagonales $\tilde{\mathbf{s}}_d$), y la entropía de Shannon en nuestros términos se puede calcular, como [9]

$$H(\tilde{\mathbf{s}}) = -\sum_{j=-n}^{n} p(\tilde{s}_j)\cdot \log_2 p(\tilde{s}_j) \tag{19}$$

donde $\tilde{\mathbf{s}}$ es la imagen transformada, s_j es el coeficiente que tiene valor j; $-n \le j \le n$ y n es el valor más grande de los coeficientes enteros resultantes de la transformada wavelet lifting.

Ya con el vector característico y los coeficientes de los filtros se va generando la memoria **M** para su aprendizaje. La fase de aprendizaje se muestra de forma general en la figura 1, donde **x** representa a los patrones espectrales de las imágenes y **y** a los coeficientes *a*, *b* correspondientes. En esta fase es donde se va asociando los coeficientes obtenidos de forma empírica y los patrones obtenidos con el modelo propuesto. El diagrama de la fase de recuperación se muestra en la figura 2.

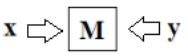


Fig. 1. Diagrama a bloques de la fase de aprendizaje.

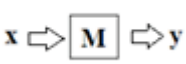


Fig. 2. Diagrama a bloques de la fase de recuperación.

En esta fase es donde se va clasificando los patrones que se le presentan a la memoria **M** para obtener de forma automática los coeficientes de los filtros wavelets, para su empleo en la compresión de imágenes.

V. Resultados

En este capítulo se presentan los resultados obtenidos de aplicar el método anteriormente descrito, los experimentos que se desarrollaron fueron usando un conjunto fundamental de 30 imágenes, algunas de ellas se muestran en la figura 3. Cabe mencionar que las imágenes son de diferentes tamaños como por ejemplo 2048 × 2560, 1524 × 1200, 1465 × 1999, 1024 × 1024 y 512 × 512 pixeles, y todas son en escala de gris de 8 bits por pixel.

Fig. 3. Imágenes naturales de prueba: Tiffany, baboon, peppers, man, couple, Lenna, sailboat, gold, f-16, Tiffany, hotel, tools.

Dentro del conjunto fundamental de imágenes también se usaron imágenes artificiales o con cierto retoque como se muestra la figura 4.

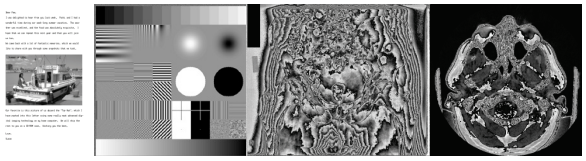


Fig. 4. Imágenes artificiales de prueba: cmpnd1, target, x_ray, ct.

Para la clasificación se usó la memoria generada **M**, y para poder comparar los resultados con esta memoria se realizaron los experimentos utilizando el clasificador *1-NN*. Los resultados obtenidos muestran que con este clasificador las imágenes de prueba dentro del conjunto fundamental presentaron un 100% de rendimiento, es decir se clasificaron correctamente el total de las imágenes del conjunto fundamental.

Para realizar un estudio comparativo de los resultados obtenidos contra los métodos ya establecidos, se utiliza un comparativo de la entropía de Shannon entre las imágenes originales (del conjunto fundamental) contra la entropía obtenida con el método presentado. Utilizando la técnica de Leave-One-Out (LOO) [31] para la validación del conjunto

ISSN 2395-8618

TABLA I
RESULTADOS DE COMPRESIÓN DE IMÁGENES DE PRUEBA (BITS/PIXEL). LOS MEJORES RESULTADOS SE MARCAN CON LA LETRA NEGRITA.

Imagen	Entropía sin transformar	Entropía obtenida con coeficientes óptimos	Entropía obtenida con LOO
aerial	6.99399	**5.2218**	5.22509
aerial2	7.19468	**5.30375**	5.30783
baboon	7.35771	**6.08492**	6.08622
baloon	7.3459	**3.01519**	3.01751
barb2	7.4838	**4.96838**	4.96945
bike	7.02187	**4.81405**	4.81733
board	6.82801	**3.87676**	3.87741
boats	7.08812	**4.18111**	4.1866
café	7.56127	**5.61969**	5.6231
couple	7.05721	**4.88488**	4.88529
ct	6.68433	**5.12178**	5.28753
Elaine	7.50598	**4.86013**	4.86058
f-16	6.70425	**4.13115**	4.13545
finger	7.4046	**5.3215**	5.33779
girl	7.28777	**3.9699**	3.97926
gold	7.52999	**4.67202**	4.67175
hotel	7.54613	**4.69236**	4.69881
Lenna	7.44743	**4.28734**	4.30198
man	7.19259	**4.71089**	4.71466
peppers	7.59427	**4.5944**	4.60044
sailboat	7.4847	**5.14825**	5.15003
target	6.36706	**3.23725**	3.34112
Tiffany	6.60019	**4.27142**	4.27186
tools	7.60137	**5.62782**	5.63006
txtur2	6.91669	**5.50499**	5.50706
water	6.8685	**3.42261**	3.42489
Tiffany	7.25149	**4.72139**	4.72376
x_ray	7.95718	**6.98739**	7.00973
Zelda	7.33354	**3.80353**	3.80497
Promedio	7.21912	**4.72609**	4.73957

TABLA II
RESULTADOS DE EXPERIMENTOS CON CDF (2,2) VS TÉCNICA PROPUESTA (BITS/PIXEL). LOS MEJORES RESULTADOS SE MARCAN CON LA LETRA NEGRITA.

Imagen	CDF (2,2)	Entropía obtenida	Entropía obtenida con LOO
aerial	5.27382	**5.2218**	5.22509
aerial2	5.33433	**5.30375**	5.30783
baboon	6.11119	**6.08492**	6.08622
baloon	3.02992	**3.01519**	3.01751
barb2	5.13575	**4.96838**	4.96945
bike	4.81578	**4.81405**	4.81733
board	3.8856	**3.87676**	3.87741
boats	4.22419	**4.18111**	4.1866
café	5.62476	**5.61969**	5.6231
couple	4.89369	**4.88488**	4.88529
Elaine	4.88923	**4.86013**	4.86058
f-16	4.16842	**4.13115**	4.13545
finger	5.49482	**5.3215**	5.33779
girl	4.07426	**3.9699**	3.97926
gold	4.67736	**4.67202**	4.67175
hotel	4.70622	**4.69236**	4.69881
Lenna	4.33814	**4.28734**	4.30198
man	4.73744	**4.71089**	4.71466
peppers	4.60908	**4.5944**	4.60044
sailboat	5.1793	**5.14825**	5.15003
Tiffany	4.28675	**4.27142**	4.27186
tools	5.64493	**5.62782**	5.63006
txtur2	5.51308	**5.50499**	5.50706
woman	4.75623	**4.72139**	4.72376
Zelda	3.84339	**3.80353**	3.80497
Promedio	4.76990	**4.73150**	4.73537

fundamental se obtuvieron los resultados que se muestran en la tabla comparativa número 1, donde se presenta los resultados de entropía de la imagen sin transformar y la entropía obtenida con los coeficientes óptimos así como con la técnica LOO para cada imagen del conjunto fundamental.

Con estos resultados se procedió a realizar el proceso de compresión del conjunto fundamental y la obtención de la entropía de cada imagen, para su comparativo contra la entropía obtenida por los filtros estándares de la familia de wavelets CDF. En la tabla 2 se realiza un comparativo del filtro CDF (2,2) contra la entropía de imágenes comprimidas con el modelo propuesto. Se puede ver que en un 100% de mejoría del método propuesto con respecto a los resultados obtenidos con el filtro CDF (2,2) y el promedio total del conjunto fundamental es menor con el método presentado.

En la tabla 3 se muestra el estudio comparativo con el filtro CDF (4,4) y el método presentado. Se puede ver que el estudio comparativo se tiene un 100% de mejora contra los resultados obtenidos por el filtro clásico CDF (4,4), esta mejora es para cada imagen del conjunto fundamental así como también se mejora la entropía promedio del conjunto general de las imágenes.

Como se puede apreciar, el método propuesto es competitivo contra los métodos ya establecidos dentro de la literatura en el caso de imágenes naturales usando coeficientes óptimos globales para cada imagen.

VI. CONCLUSIONES

Se desarrolló un modelo de generación automática de coeficientes para filtros wavelets analizando las propiedades espectrales de las imágenes en el dominio de la transformada DCT y usando modelos de inteligencia artificial, específicamente, *1*-NN, el cual se demostró ser competitivo contra los modelos ya establecidos dentro de la bibliografía.

Se probó el algoritmo desarrollado con diferentes datos de imágenes para la compresión sin pérdidas y se comparó con los resultados obtenidos con las técnicas existentes, encontrándose y demostrándose las bondades de la aportación descrita en el presente trabajo. Se obtuvo una mejor compresión de las diferentes imágenes con relación a la

ISSN 2395-8618

TABLA III
RESULTADOS DE EXPERIMENTOS CON CDF (4,4) VS TÉCNICA PROPUESTA (BITS/PIXEL). LOS MEJORES RESULTADOS SE MARCAN CON LA LETRA NEGRITA.

Imagen	CDF (4,4)	Entropía obtenida	Entropía obtenida con LOO
aerial	5.22282	**5.2218**	5.22509
aerial2	5.30469	**5.30375**	5.30783
baboon	6.08569	**6.08492**	6.08622
baloon	3.01551	**3.01519**	3.01751
barb2	5.08559	**4.96838**	4.96945
bike	4.81534	**4.81405**	4.81733
board	3.87838	**3.87676**	3.87741
boats	4.18232	**4.18111**	4.1866
café	5.6231	**5.61969**	5.6231
cmpnd1	3.34527	**2.925**	3.3135
couple	4.61841	**4.88488**	4.88529
ct	5.33138	**5.12178**	5.28753
Elaine	4.8613	**4.86013**	4.86058
f-16	4.13125	**4.13115**	4.13545
finger	5.354	**5.3215**	5.33779
girl	3.97926	**3.9699**	3.97926
gold	4.6738	**4.67202**	4.67175
hotel	4.69675	**4.69236**	4.69881
Lenna	4.28893	**4.28734**	4.30198
man	4.71113	**4.71089**	4.71466
peppers	4.59882	**4.5944**	4.60044
sailboat	5.14887	**5.14825**	5.15003
target	3.36225	**3.23725**	3.34112
Tiffany	4.27145	**4.27142**	4.27186
tools	5.62856	**5.62782**	5.63006
txtur2	5.50565	**5.50499**	5.50706
water	3.42768	**3.42261**	3.42489
woman	4.72292	**4.72139**	4.72376
x_ray	6.97004	**6.98739**	7.00973
Zelda	3.83507	**3.80353**	3.80497
Promedio	4.68533	**4.66605**	4.69203

entropía obtenida, que es competitiva contra los filtros de la familia CDF(2,2) y CDF(4,4) en imágenes naturales.

Cabe señalar que el método propuesto utiliza los coeficientes *a,b* óptimos globales para cada imagen; por otra parte, se pueden encontrar los coeficientes *a,b* óptimos para cada cuadrante de descomposición wavelets, y con ello realizar clasificación a nivel de cada cuadrante de descomposiciones para obtener así el nivel de compresión aún más alto. En mismo tiempo, la mejor técnica para algunas imágenes artificiales fue CDF(2,2), que significa que el modelo propuesto debe de tener la entrada de un parámetro de "artificialidad en los datos" para una mejor clasificación de coeficientes *a,b*. Estas mejoras pueden ser el objetivo del trabajo futuro.

AGRADECIMIENTOS

El primer autor agradece al CONACyT por la beca otorgada durante los estudios doctorales. Este trabajo fue parcialmente apoyado por Instituto Politécnico Nacional con el proyecto de investigación SIP 20161173.

REFERENCIAS

[1] I. Daubechies, "Orthogonal bases of compactly supported wavelets," *Comm. Pure Appl. Math,* vol. 41, pp. 909–996, 1988..

[2] A. Cohen, I. Daubechies y J. Feauveau, "Bi-orthogonal bases of compactly supported wavelets," *Comm. Pure Appl. Math,* vol. 45, pp. 485–560, 1992.

[3] A. H. Tewfik, D. Sinha y P. Jorgensen, "On the optimal choice of a wavelet for signal representation," *IEEE Trans. Inform.Theory,* vol. 38, p. 747–765, 1992.

[4] S. G. Mallat y Z. Zhang, "Matching pursuit with time-frequency dictionaries," *IEEE trans. Signal Processing,* vol. 41, pp. 3397–3415, 1993.

[5] A. Aldroubi y M. Unser, "Families of multiresolution and wavelet spaces with optimal properties," *Numer Func. Anal.,* vol. 14, nº 5/6, pp. 417–446, 1993.

[6] J. O. Chapa y R. M. Raghuveer, "Algorithms for Designing Wavelets to Match a Specified Signal," *IEEE Trans. Signal Processing,* vol. 48, nº 12, pp. 3395–3406, 2000.

[7] A. Gupta, S. Dutt Joshi y S. Prasad, "A New Approach for Estimation of Statistically Matched Wavelet," *IEEE Trans. Signal Processing,* vol. 53, nº 5, pp. 1778–1793, 2005.

[8] W. Sweldens, "The lifting scheme: A new philosophy in biorthogonal wavelet constructions," *Wavelet Applications in Signal and Image Processing III,* pp. 68–79, 1995.

[9] R. Calderbank, I. Daubechies, W. Sweldens y B. L. Yeo, "Wavelet transforms that maps integers to integers," *Appl. Comput. Harmon. Anal,* vol. 5, nº 3, pp. 332–369, 1996..

[10] I. Daubechies y W. Sweldens, "Factoring Wavelet Transforms Into Lifting Steps," *J. Fourier Anal. Appl,* vol. 4, nº 3, pp. 245–267, 1998.

[11] N. V. Boulgouris y M. G. Strintzis, "Lossless Image Compression Based on Optimal Prediction, Adaptive Lifting and Conditional Arithmetic Coding," *IEEE Transactions on Image Processing,* vol. 10, pp. 1–14, 2001.

[12] M. Kaaniche, B. Pesquet-Popesku, A. Benazza-Benyhahia and J-C Resquet. "Adaptive lifting scheme with sparse criteria for image coding," EURASIP Journal on Advances in Signal Processing, Vol. 2012, No. 1, p.p. 1–12, 2012..

[13] H. Thielemann, "Optimally matched wavelets Ph.D thesis," *Universität Bremen, Vorgelegt im Fachbereich 3 (Mathematik und Informatik),* 2005.

[14] H. Li, G. Liu y Z. Zhang, "Optimization of Integer Wavelet Transforms Based on Difference Correlation Structures," *IEEE Trans Image Processing,* vol. 14, nº 11, pp. 1831–1847, 2005.

[15] Kitanovski, M. Kseneman, D. Gleich y D. Taskovski, "Adaptive Lifting Integer Wavelet Transform for Lossless Image Compression," *Proc. of 15th International Conference on Systems, Signals and Image Processing UWSSIP 2008,* pp. 105–108, 2008.

[16] S. G. Mallat , "Multifrequency channel decompositions of images and wavelet models," *IEEE Trans. Acoust. Speech Signal Process,* vol. 37, nº 12, pp. 2091–2110, 1989.

[17] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE transactions on pattern analysis and machine intelligence,* vol. 11, nº 7, pp. 674–693, 1989.

[18] M. Vetterli, "Multi-dimensional sub-band coding: some theory and algorithms," *Signal Processing,* vol. 6, pp. 97–112, 1984.

[19] W. Sweldens, "The Lifting Scheme: A Construction of Second Generation Wavelets," *Siam J. Math. Anal,* vol. 29, nº 2, pp. 511–546, 1997.

[20] J. G. Proakis y D. G. Manoakis, Tratamiento digital de señales, 3ra. ed., Madrid: Prentice Hall Int., 1998.

[21] H. Yoo y J. Jeong, "A unified framework for wavelet transforms based on the lifting scheme," *IEEE Conference Publications,* vol. 3, pp. 792–795, 2001.

ISSN 2395-8618

[22] E. Fix y J. L. Hodges, "Discriminatory analysis, nonparametric discrimination," *USAF School of Aviation Medivine, Randolph Field,* 1951.

[23] T. M. Cover y P. E. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory,* vol. 13, nº 1, pp. 21–27, 1967.

[24] S. Hotta, S. Kiyasu y S. Miyahara, "Pattern Recognition Using Average Patterns of Categorical k-Nearest Neighbors," *Proceedings of the 17th International Conference on Pattern Recognition,* vol. 4, pp. 412–415, 2004.

[25] N. El Gayar, F. Schwenker y G. Palm, "A Study of the Robustness of KNN Classifiers Trained Using Soft Labels," *Springer-Verlag ANNPR 2006,* pp. 67–80, 2006.

[26] Y. Song, J. Huang, D. Zhou, H. Zha y C. L. Giles, "IKNN: Informative K-Nearest Neighbor Pattern Classification," *Springer-Verlag,* p. 248–264, 2007.

[27] N. Peng, Y. Zhang y Y. Zhao, "A SVM-kNN method for quasar-star classification," *Sci China-Phys Mech Astron,* vol. 56, nº 6, pp. 1227–1234, 2013.

[28] A. Dhurandhar y A. Dobra, "Probabilistic characterization of nearest neighbor classifier," *Int. J. Mach. Learn. & Cyber,* vol. 4, p. 259–272, 2013.

[29] J. M. Shapiro, "Embedded Image Coding Using Zerotrees Of Wavelet Coefficients," IEEE Trans Signal Processing, Vol. 41, No. 12, p.p. 3445–3462, Dec. 1993.

[30] David J.C. MacKay. Information Theory, Inference, and Learning Algorithms. Cambridge University Press, 2003.

[31] R. O. Duda, P. E. Hart y D. G. Stork , Pattern Classification., 2da ed., John Wiley & Sons, 1997.

ISSN 2395-8618

Data Reduction and Regression Using Principal Component Analysis in Qualitative Spatial Reasoning and Health Informatics

Chaman Lal Sabharwal and Bushra Anjum

***Abstract*—The central idea of principal component analysis (PCA) is to reduce the dimensionality of a dataset consisting of a large number of interrelated variables, while retaining as much as possible of the variation present in the dataset. In this paper, we use PCA based algorithms in two diverse genres, qualitative spatial reasoning (QSR) to achieve lossless data reduction and health informatics to achieve data reduction along with improved regression analysis respectively. In an adaptive hybrid approach, we have employed PCA to traditional regression algorithms to improve their performance and representation. This yields prediction models that have both a better fit and reduced number of attributes than those produced by using standard logistic regression alone. We present examples using both synthetic data and real health datasets from UCI Repository.**

***Index Terms*—Principal component analysis, regression analysis, healthcare analytics, big data analytics, region connection calculus.**

I. Introduction

PCA is a way of identifying patterns in data, and expressing the data in such a way as to highlight their similarities and differences. Since patterns in data can be hard to find in data of high dimension, where the luxury of graphical representation is not available, PCA is a powerful tool for analyzing data. The other main advantage of PCA is that once we have found these patterns in the data, then we compress the data, i.e., by reducing the number of dimensions, without much loss of information.

PCA and Singular Value Decomposition (SVD) are interchangeably used for data reduction/compressions whereas statistical techniques such as regression analysis are used for approximation and analysis of data. Such applications include data mining, health informatics, oceanography, meteorology, natural language processing, machine learning, image analysis, geometry visualization.

Manuscript received on January 25, 2016, accepted for publication on May 20, 2016, published on June 25, 2016.

Chaman Lal Sabharwal is with the Missouri University of Science and Technology, Rolla, MO-63128, USA (e-mail: chaman@mst.edu).

Bushra Anjum is with Amazon Inc,, 1194 Pacific St., San Luis Obispo, CA-93401, USA (e-mail: banjum@amazon.com).

A. Qualitative Spatial Reasoning

Reasoning about spatial data is a key task in many applications, including geographic information systems, meteorological and fluid flow analysis, computer-aided design, and protein structure databases. Such applications often require the identification and manipulation of qualitative spatial representations, for example, to detect whether one "object" will soon occlude another in a digital image, or to determine efficiently relationships between a proposed road and wetland regions in a geographic dataset. QSR provides representational primitives (a spatial "vocabulary") and inference mechanisms.

Much QSR work has studied purely topological descriptions of spatial regions and their relationships. One representative approach, the Region-Connection Calculus (RCC), provides predicates for expressing and reasoning about the relationships among topological regions (arbitrarily shaped chunks of space). RCC was originally designed for 2D [1, 2]; later it was extended to 3D [3]. Herein we introduce PCA to reduce 9-Intersection model to 4-intersection model in both 2D and 3D. The performance of QSR can be improved by reducing the number of intersections, but PCA connection to QSR is non-existent in the literature. Herein we show how (1) PCA can be applied to intersection dimension reduction for QSR spatial data, and (2) the 9-Intersection can be reduced to 4-Intersection for all spatial as well as non-spatial objects.

For example, of item-attribute-concept in RCC, spatial objects are items, their intersections are attributes, and relations are concepts. There are five RCC5 and eight RCC8 concepts in RCC, see Figure 1, and [2, 3].

B. Health Informatics

The Big Data revolution has begun for many industries. The healthcare industry has been playing catch up and has finally reached a consensus on the value of Big Data as a transformative tool. Statistical linear and logistic regression that have been the popular mining techniques, but their ability to deal with inter dependent factors is limited. The understanding of principal components, however, has been

ISSN 2395-8618

lacking in the past by non-academic clinicians. It is no surprise that keeping people healthy is costing more money. From the price of medications and the cost of hospital stays to doctors' fees and medical tests, health-care costs around the world are skyrocketing. Much of this is attributed to wasteful spending on such things as ineffective drugs, futile procedures and redundant paperwork, as well as missed disease-prevention opportunities. This calls for mechanism for efficient data reduction and diagnostic tools as pointed out by some of the examples in the literature cited in the next paragraph.

Analysis of this Big Data offers unlimited opportunities for healthcare researchers and it is estimated that developing and using prediction models in the health-care industry could save billions by using big-data health analytics to mine the treasure trove of information in electronic health records, insurance claims, prescription orders, clinical studies, government reports, and laboratory results. According to the Harvard School of Public Health publication entitled The Promise of Big Data, petabytes of raw information could provide clues for everything from preventing tuberculosis to shrinking health care costs—if we can figure out how to apply this data [4]. Improving the care of chronic diseases, uncovering the clinical effectiveness of treatments, and reducing readmissions are expected to be top priority use cases for Big Data in healthcare [5].

In this paper, we will give general guidelines to address various issues. We explore an adaptive hybrid approach (1) how PCA can be used to reduce data in the original space in addition to transformed space, (2) how PCA can be used to improve standard line regression and logistic regression algorithms, (3) how to use logistic regression in conjunction PCA to yield models which have both a better fit and reduced number of variables than those produced by using logistic regression alone.

The paper is organized as follows. Section II describes the background on linear regression, logistic regression, and principle of component analysis. Section III describes PCA in detail, along with our suggested representational improvements. Section IV presents the hybrid algorithms for regression using PCA. Section V discusses PCA's improved role in dimensionality reduction followed by additional experimental support in Section VI. Section VII concludes the paper.

II. Background

A. *Mathematical Notation*

In this section, we describe the mathematical notation for terms whose definitions will follow in the paper. A vector is a sequence of elements. All vectors are *column* vectors and are in lower case *bold* letters such as **x**. The n-tuple $[x_1,..., x_n]$ denotes a *row* vector with n elements in lowercase. A superscript T is used to denote the *transpose* of a vector $\boldsymbol{x}$, so that $\boldsymbol{x}^T$ is a row vector whereas $\boldsymbol{x} = [x_1,..., x_n]^T$ is a column vector. This notation is overloaded at some places where the ordered pair $[x_1, x_2]$ may be used as a row *vector*, a *point* in the plane or a closed *interval* on the real line. The *matrices* are denoted with uppercase letters, e.g. A, B. For vectors $\boldsymbol{x}, \boldsymbol{y}$, the covariance is denoted by $\text{cov}(\boldsymbol{x}, \boldsymbol{y})$, whereas $\text{cov}(\boldsymbol{x})$ is used for $\text{cov}(\boldsymbol{x}, \boldsymbol{x})$ as a shortcut [6].

If we have m vector values $\boldsymbol{x}_1,...,\boldsymbol{x}_m$ of an n-dimensional vector $\boldsymbol{x} = [x_1,..., x_n]^T$, these m row vectors are collectively represented by an $m \times n$ data matrix A**.** The k^{th} row of A is the row vector $\boldsymbol{x}_k^T$. Thus the (i, j) element of A becomes the j^{th} element of the i^{th} row/observation, $\boldsymbol{x}_i^T$.

There are several ways to represent data so that implicit information becomes explicit. For linear representation of vector data, a vector space is equipped with a basis of linearly independent vectors. Usually in data mining, the data is represented as a matrix of row vectors or data instances. Two of the methods for efficient representation of data are regression and PCA.

B. *Qualitative Spatial Reasoning*

Much of the foundational research on QSR is related to RCC that describes two regions by their possible relations to each other. RCC5/RCC8 can be formalized by using first order logic [2] or by using the 9-intersection model [1]. Conceptually, for any two regions, there are three possibilities: (1) *one object is outside the other*; this results in the RCC5 relation DR (interiors disjoint) and RCC8 relation DC (disconnected) or EC (externally connected). (2) *One object overlaps the other across boundaries*; this corresponds to the RCC5/RCC8 relation PO (proper overlap). (3) *One object is inside the other*; this results in topological relation EQ (equal) or RCC5 relation PP (proper part). To make the relations jointly exhaustive and pairwise distinct (JEPD), there is a converse relation denoted by PPc (proper part *converse*), PPc(A,B) ≡ PP(B,A). For a close examination, RCC8 decomposes RCC5 relation PP (proper part) into two relations: TPP (tangential proper part) and NTPP (non-tangential Proper part). Similarly for RCC5 relation PPc, RCC8 defines TPPc and NTPPc. The RCC5 and RCC8 relations are pictorially described in Figure 1.

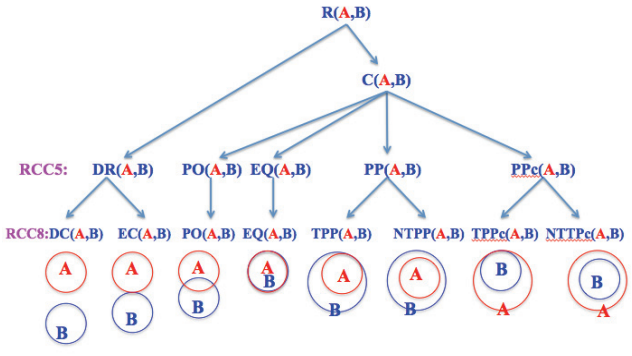


Fig. 1. RCC5 and RCC8 relations in 2D

ISSN 2395-8618

Each of the RCC8 relations can be uniquely described by using the 9-Intersection framework. It is a comprehensive way to look at any relation between two regions. The 9-Intersection matrix for two regions *A* and *B* is given in Table 1, where Int represents the region's interior, Bnd denotes the boundary, and Ext represents the exterior. The predicate IntInt(*A,B*) is a binary relation that represents the intersection between the interiors of region *A* and region *B*; the value of this function is either true (non-empty) or false (empty). Similarly, there are other predicates for the intersection of *A*'s interior, exterior, or boundary with those of *B*. In QSR, we are only concerned with the presence/absence of an intersection. The actual value of the intersection is not necessary.

For two non-empty bounded regions *A* and *B*, the intersection of their exteriors is always non-empty. This is represented in the last column of Table 2. Since it adds no new information, it has been proposed in the literature [2] to replace the 9-Intersection with the 8-Intersection model to define the spatial relations. The values of the 8-Intersection framework for the RCC8 framework are given in the first eight columns in Table 2.

TABLE 1.
9-INTERSECTION MATRIX FOR CALCULATING RCC8 RELATIONS

	Interior	Boundary	Exterior
Interior	Int(A)∩Int(B)	Int(A)∩Bnd(B)	Int(A)∩Ext(B)
Boundary	Bnd(A)∩Int(B)	Bnd(A)∩Bnd(B)	Bnd(A)∩Ext(B)
Exterior	Ext(A)∩Int(B)	Ext(A)∩Bnd(B)	Ext(A)∩Ext(B)

TABLE II
BOOLEAN VALUES FOR 9-INTERSECTIONS REQUIRED TO DISTINGUISH EACH RCC8 RELATION

RCC8	Int Int	Bnd Bnd	Bnd Int	Int Bnd	Int Ext	Bnd Ext	Ext Int	Ext Bnd	Ext Ext
DC	F	F	F	F	T	T	T	T	**T**
EC	F	T	F	F	T	T	T	T	**T**
EQ	T	T	F	F	F	F	F	F	**T**
NTPPc	T	F	F	T	T	T	F	F	**T**
TPPc	T	T	F	T	T	T	F	F	**T**
NTPP	T	F	T	F	F	F	T	T	**T**
TPP	T	T	T	F	F	F	T	T	**T**
PO	T	T	T	T	T	T	T	T	**T**

In this paper, we show that PCA provides a better alternative to conventional methods of dimensionality reduction in QSR. The analysis is equally applicable to both non-spatial discrete web objects as well as conventional spatial objects such as cuboids and spheres.

In such applications, some threshold may be required to interpret the resulting dimensions. One can simply ignore variation below a particular threshold to reduce the data and still preserve the main concepts of original intent.

C. Health Informatics

There is a significant opportunity to improve the efficiencies in the healthcare industry by using an evidence-based learning model, which can in turn be powered by Big Data analytics [7]. A few examples are provided below. The company Asthmapolis has created a global positioning system (GPS) enabled tracker that monitors inhaler usage by patients, eventually leading to more effective treatment of asthma [8]. Center for Disease Control and Prevention (CDC) is using Big Data analytics to combat influenza. Every week, the CDC receives over 700,000 flu reports including the details on the sickness, what treatment was given, and whether not the treatment was successful.

The CDC has made this information available to the general public called FluView, an application that organizes and sifts through this extremely large amount of data to create a clearer picture for doctors of how the disease is spreading across the nation in near real-time [9]. GNS Healthcare, a Big Data analytics company, has come together with the health insurance company Aetna to help combat people at risk or already with metabolic syndromes. The company has developed a technology known as Reverse Engineering and Forward Simulation that will be put to work on the data of Aetna insurance subscribers. Essentially, the technology will search for the presence of five warning signs: large waist size, high blood pressure, high triglycerides, low High density Lipoprotein, and high blood sugar. A combination of any three of these lead to the conclusion that the patient is suffering from the condition [10].

Researchers at Allazo Health are creating systems designed to improve on medication adherence programs by using predictive analytics. For example, predict what interventions are mostly likely to work for that patient based on what interventions already worked for other patients with similar demographics, behavioral profiles, and medical history [11]. Another area of interest is the surveillance of adverse drug reactions (ADRs) which has been a leading cause of *death* in the United States [12]. It is estimated that approximately 2 million patients in USA are affected by ADRs and the researchers in [13], [14] and [15] propose an analytical framework for extracting patient–reported adverse drug events from online patient forums such as DailyStrength and PatientsLikeMe.

Simplistically speaking, in all the above examples the researchers are trying to model and predict a dependent phenomenon based on a number of predictors that have been observed. The dependent parameter can be discrete, nominal, or even binary / logical. There are two problems at hand: dimension reduction and prediction. First problem is data optimization. The optimization problem is data cleaning, and how we can reduce the set of predictors while still maintaining a high prediction accuracy for the dependent variable. The second problem is the prediction of the dependent variable from the reduced dataset. This problem is

ISSN 2395-8618

analyzing whether some event occurred or not given the success or failure, acceptance or rejection, presence or absence of observed simulators. This is where PCA comes into the picture.

III. Principle Component Analysis

The PCA is a well-known data reduction tool in academia for over 100 years. PCA creates a linear orthogonal transformation of correlated data in one frame (coordinates system) to uncorrelated data in another frame. The huge dimensional data can be transformed and approximated with a few dimensions. PCA finds the directions of maximum variance in high-dimensional data and projects it onto a smaller dimensional subspace while retaining most of the original information. If the data is noisy, PCA reduces noise implicitly while projecting data along the principal components. In this paper, we explore an adaptive hybrid approach to show that PCA can be used not only for data reduction but also for regression algorithm improvement. We will describe the hybrid model for both linear and logistic regression algorithms.

Before delving further, we would like to discuss the terms PCA and SVD further as they are used interchangeably in the literature. There is a clear distinction between them.

Definition 1. For a real square matrix A, if there is a real number λ and a *non-zero* vector $\boldsymbol{x}$ such that $A\,\boldsymbol{x} = \lambda\,\boldsymbol{x}$, then λ is called an eigenvalue and $\mathbf{x}$ is called an eigenvector.

Definition 2. For a real matrix A (square or rectangular), if there a *non-negative* real number σ and a non-zero vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ such that $A^T\,\boldsymbol{x} = \sigma\,\boldsymbol{y}$, and $A\,\boldsymbol{y} = \sigma\,\boldsymbol{x}$, then σ is called a singular value and $\boldsymbol{x}$ and $\boldsymbol{y}$ represent a pair of singular vectors [16].

Note 1. λ can be negative or positive, but σ is always non-negative.

Note 2. σ^2 is an eigenvalue of covariance matrices AA^T and A^TA. This can be quickly seen

$$A^T\boldsymbol{x} = \sigma\boldsymbol{y} \rightarrow AA^T\boldsymbol{x} = \sigma A\boldsymbol{y} \rightarrow AA^T\boldsymbol{x} = \sigma\sigma\boldsymbol{x} = \sigma^2\boldsymbol{x}$$

Therefore

$$AA^T\boldsymbol{x} = \sigma^2\boldsymbol{x}$$

Similarly, we can see that

$$A^TA\,\boldsymbol{y} = \sigma^2\boldsymbol{y}$$

An eigenvector is a direction vector supporting the spread of data along the direction of the vector. An eigenvalue measures the spread of data in the direction of the eigenvector. Technically, a principal component can be defined as a linear combination of optimally weighted observed variables. The words "linear combination" refer to the fact that weights/coefficients in a component are created by the contribution of the observed variables being analyzed. "Optimally weighted" refers to the fact that the observed variables are weighted in such a way that the resulting components account for a maximal amount of variance in the dataset.

This will also be a good place to introduce Least Square Approximation (LSA). LSA and PCA are both linear transformations. However, they accomplish the same task differently. In a vector space, for any vector $\boldsymbol{v}$ and a unit vector $\boldsymbol{u}$, we have $\boldsymbol{v} = \boldsymbol{v}\bullet\boldsymbol{u}\,\boldsymbol{u} + (\boldsymbol{v} - \boldsymbol{v}\bullet\boldsymbol{u}\,\boldsymbol{u})$. Finding the vector $\boldsymbol{u}$ that minimizes $|(\boldsymbol{v} - \boldsymbol{v}\bullet\boldsymbol{u}\,\boldsymbol{u})|$ is the same as finding a vector $\boldsymbol{u}$ that maximizes $|\boldsymbol{v}\bullet\boldsymbol{u}|$. LSA calculates the direction $\boldsymbol{u}$ that minimizes the variance of data *from* the direction $\boldsymbol{u}$ **whereas** PCA computes the direction $\boldsymbol{u}$ (principal component) that maximizes the variance of the data *along* the direction $\boldsymbol{u}$. This concept is applied to all data instance vectors collectively resulting in covariance matrix AA^T of data matrix and $\boldsymbol{u}$ *is* the eigenvector of AA^T with largest eigenvalue.

If A is a real square symmetric matrix, then eigenvalues are real and eigenvectors are orthogonal [17]. PCA computes eigenvalues and eigenvectors of a data matrix to project data on a lower dimensional subspace. PCA decomposition for a square symmetric matrix A is $A = UDU^T$ where U is the matrix of eigenvectors and D is diagonal matrix of eigenvalues of A. Since $AU = UD$, U is orthogonal, therefore $A=UDU^T$. Also PCA orders the eigenvalues in the descending order of magnitude. The columns of U and diagonal entries of D are arranged correspondingly. Since eigenvalues can be negative, the diagonal entries of D are ordered based on absolute values of eigenvalues.

The SVD decomposition is applicable to a matrix of *any size* (not necessarily square and symmetric). $A = USV^T$, where U is the matrix of eigenvectors of covariance matrix AA^T, V is the matrix of eigenvectors of covariance matrix A^TA, and S is a diagonal matrix with eigenvalues as the main diagonal entries. Hence, PCA can use SVD to calculate eigenvalues and eigenvectors. Also SVD calculates U and V efficiently by recognizing that if $\mathbf{v}$ is an eigenvector of A^TA for non-zero eigenvalue λ, then $A\mathbf{v}$ is automatically an eigenvector of AA^T for the same eigenvalue λ where $\lambda \geq 0$. If $A\mathbf{v}$ is an eigenvector, say, $\mathbf{u}$, then $A\mathbf{v}$ is a multiple of eigenvector $\mathbf{u}$ (since $\mathbf{u}$ and $\mathbf{v}$ are unit vectors) and it turns out that $A\boldsymbol{v} = \sqrt{\lambda}\boldsymbol{u}$, or $\boldsymbol{u} = \sigma A\boldsymbol{v}$, where $\sigma = 1/\sqrt{\lambda}$. By convention, SVD ranks the eigenvectors on descending order of eigenvalues. If U and V are matrices of eigenvectors of AA^T and A^TA, and S is the matrix of square roots of eigenvalues on the main diagonal, then A can be expressed as $A = USV^T$ [17]. The eigenvalues of a square symmetric matrix A are square roots of eigenvalues of AA^T. We will show that it is sufficient to have S as diagonal matrix of only non-zero eigenvalues and U, V to have columns of only the corresponding eigenvectors.

In data mining, the m observations/data points are represented as an $m \times n$ matrix A where each observation is a vector with n components. PCA/SVD help in transforming

ISSN 2395-8618

physical world data / objects more clearly in terms of independent, uncorrelated, orthogonal parameters.

The first component extracted in principal component analysis accounts for a maximal amount of variance in the observed variables. The second component extracted will have two important characteristics. First, this component will account for a maximal amount of variance in the dataset that was not accounted for by the first component. The second characteristic of the second component is that it will be uncorrelated with the first component. The remaining components that are extracted in the analysis display the same two characteristics. Visualizing graphically, for the first component direction (eigenvector) e_1, the data spread is maximum m_1; for the next component direction e_2, the data spread is next maximum m_2 ($m_2 < m_1$ the previous maximum) and the direction e_2 is orthogonal to previous direction e_1.

Note 3. For the eigenvectors $\boldsymbol{u}$ of A, any non-zero multiple of $\boldsymbol{u}$ is also an eigenvector of A. In U, the eigenvectors are normalized to unity. If $\boldsymbol{u}$ is a unit eigenvector, then $-\boldsymbol{u}$ is also a unit eigenvector. Thus the sign can be arbitrarily chosen. Some authors make the first nonzero element of the vector to be positive to make them unique. We do not follow this convention. Since the eigenvectors are ordered, we make the k^{th} element of vector $\boldsymbol{u}_k$ positive. If k^{th} element is zero, only then first non-zero element is made positive. This is a better representation of eigenvectors as it represents the data in a right-handed system as opposed to asymmetrical ordering, see Figure 2. Using Matlab svd(A) on a simple set of only two data points, the algorithm generates two orthogonal vectors v_1, v_2 as given in Figure 2(a, b). Our algorithm finds green vectors in Figure 2(c), which is more natural and of a right-handed orientation.

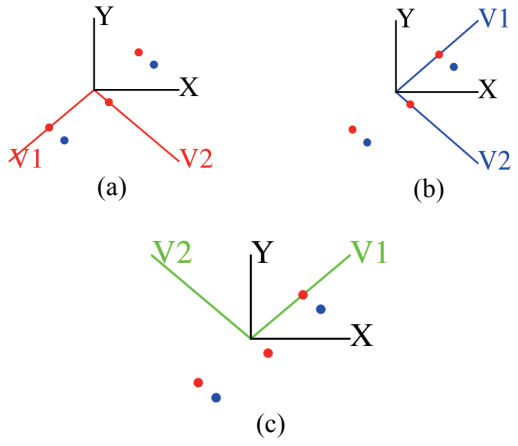


Fig. 2: Gray set of axes are standard xy-system. Blue dots represent a set of data points. Red dots are projections of data points on the principal components. (a) The axes v1, v2 in red are the eigenvectors that are computed by Matlab using the set of data points. (b) The axes v1, v2 in blue are the directions so that each eigenvector is unique making the first non-zero component positive, used in the literature. (c) The axes v1, v2 in green are generated when the sign in an eigenvector is chosen by our scheme.

A. Properties of PCA/SVD

In essence, PCA/SVD takes a high dimensional set of data points and reduces it to a lower dimensional space that exposes the knowledge that is hidden in the original space. Moreover, the transformation makes similar items more similar, and dissimilar items more dissimilar.

It is not the goal of SVD to get back the original matrix. However, the reduced dimensions do give insight about the original matrix, as we will see in the case of QSR and health informatics.

There are three important properties of SVD:

(1) It can transform correlated variables into a set of uncorrelated ones. SVD computes a transformation T such covariance of TA is diagonal matrix D, e.g. $TA(TA)^T$ is D. It can extract some relationships hidden in the original data. It can also reduce noise in the data [18].

(2) Since the eigenvectors are ordered on most variation to least variation in descending order of eigenvalues, deleting the trailing eigenvectors of smaller variation ensures minimal error. It finds the best *approximation* of the original data points by projecting data on a fewer dimensional subspace; see Figure 3 of linear data.

(3) The data can be reduced to any desired size. The *accuracy* of smaller size data depends on the reduction in the number of dimensions [19]. By deleting eigenvectors corresponding to least variation, we effectively eliminate noise in the representation of data vectors [20].

B. Dimension Reduction using PCA

There is a multitude of situations in which data is represented as a matrix. In fact, most of the real world data is expressed in terms of vectors and matrices where each vector has a large number of attributes. Matrices are used to represent data elegantly, and efficiently. In a matrix, each column represents a conceptual attribute of all the items, and each row represents all attributes related to individual data item. For example, in health informatics, rows may represent patients and columns may represent disease diagnostic symptoms or the rows may represent medicines and columns may represent side effects or adverse reactions. Similarly, in spatial-temporal reasoning, rows represent pairs of objects and columns represent temporal intersection properties of objects.

The goal of PCA is to reduce the big data matrix A to a smaller matrix retaining approximately the same information as the original data matrix and make the knowledge explicit that was implicit in the original matrix.

Example: In this example, we have 20 three-dimensional points in 20×3 matrix A. Each row of A has three values for x-, y-, z- coordinates. Visually we can see that the data points have a linear trend, in 3D, but data points have noise components in the y, and z coordinates. PCA determines the

ISSN 2395-8618

direction and eliminates noise by eliminating the eigenvectors corresponding to smaller eigenvalues (see Figure 3). Black "+" symbols represent the original data points with noise. As can be seen visually, they are almost linear. They can be approximated along eigenvector v_1 in one dimension of the $v_1v_2v_3$-system. Red lines depict the data trend in the three dimensional space along v_1, v_2 and v_3 direction. Blue lines depict the data spread in the three dimensional space along v_1, v_2 and v_3 direction. Since the data spread or blue line on v_2 and v_3 is almost non-existent (closer to the origin), it is an indicator of noise in the linear data. We can see that the data can be represented satisfactorily using only one direction.

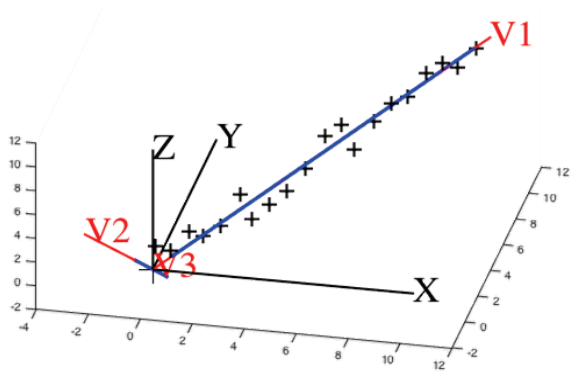


Fig. 3. Data is almost linear along v_1 direction in 3D, the noise is along the y and z-axis. Blue lines reflect data spread along v_1, v_2 and v_3 directions.

Table 3 enumerates the numeric values of the PCA decomposition. First row lists the eigenvalues, which represent the spread of points along the principal components. Second row shows the eigenvectors corresponding to the eigenvalues. The next three rows represent the error on using first, first two, first three eigenpairs. Let *newA* be the USV^T based on eigenpairs used. Then Error Original is the $|A\text{-}newA|/|A|$ percentage error in the original space. Error Projection is the $|AV\text{-}newAV|/|AV|$ percentage error in the projection space. Error Eigenvalues is the $\sum_{p=k+1,3} \lambda_p / \sum_{p=1,3} \lambda_p$ percentage error in the eigenvalue space.

Table 3 reveals that v_1, is the data dimension and v_2, v_3 correspond to the noise in this case. If we use two (or three) eigenpairs there is no error. Hence, the data can be represented in one dimension only instead of three dimensions with slight error of 2%. This error is attributed to the difference between the non-zero eigenvalues. The third and fifth rows in Table 3 indicate that the error metrics are equivalent in the original and projection space.

TABLE III
EIGENVALUES, EIGENVECTORS AND PERCENT ERRORS FOR ONE DIMENSIONAL DATA IN 3D

	Eigenvectors		
Eigenvalues	10.094419	0.213053	0
Eigenvectors	[0.6,0.6,0.6]	[-0.8,0.4,0.4]	[0.0,-0.7,0.7]
Error Original	2.110605	0	0
Error Eigenvalue	2.066979	0	0
Error Projection	2.110605	0	0

IV. PRINCIPAL COMPONENT AND REGRESSION ANALYSIS

There are several ways to model the prediction variables, e.g., linear regression analysis, logistic regression analysis, and PCA. Each has its own advantages. Though regression analysis has been well known as a statistic technique, the understanding of principles, however, has been lacking in the past by non-academic clinicians [21]. In this paper we explore an adaptive hybrid approach where PCA can be used in conjunction with regression to yield models which have both a better fit and reduced number of variables than those used by standalone regression. We will apply our findings to a medical dataset obtained from UCI Machine Learning Repository about liver patient [22]. We use the records for detecting the existence or non-existence of liver disease based on several factors such as age, gender, total bilirubin, etc.

A. *PCA and Linear Regression*

In linear regression, the distance between *observed* point (x_i,y_i) from the *computed* point $(x_i,a+bx_i)$ on line $y=a+bx$, is minimized along the y direction. In PCA, the distance of the *observed* point (x_i,y_i) is minimized to a line which is orthogonal to the line $y=a+bx$, is minimized. The details of linear algebra concepts in this section are found in [17]. We assume that data is standardized to mean zero; and normalized appropriately by the number of objects.

For one independent and one dependent variable, the regression line is $y=a+bx$ where the error between the *observed* value y_i and *estimated* value $a+bx_i$ is minimum. For n points data, we compute a and b by using the method of least squares that minimizes:

$$\sum_{i=1,n} (\boldsymbol{y_i - a - bx_i})^2$$

This is a standard technique that gives regression coefficients a and b where a is the y-intercept and b is the slope of the line.

$$\begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{bmatrix} = \frac{\begin{bmatrix} \boldsymbol{cov(x)} & \boldsymbol{-\bar{x}} \\ \boldsymbol{-\bar{x}} & \boldsymbol{1} \end{bmatrix}}{\boldsymbol{cov(x) - \bar{x}^2}} \begin{bmatrix} \boldsymbol{\bar{y}} \\ \boldsymbol{cov(x,y)} \end{bmatrix}$$

If the data is mean-centered, then a=0 because $\bar{x} = \frac{\sum_{i=1,n} x_i}{n} = 0$ and $\bar{y} = \frac{\sum_{i=1,n} y_i}{n} = 0$. Alternatively, we can replace x_i with $x_i - \bar{x}$ and y_i with $y_i - \bar{y}$. The direction of the line is always obtained from b as $\left[\frac{1}{\sqrt{1+b^2}}, \frac{b}{\sqrt{1+b^2}}\right]$.

For more than one independent variables, say m, we have

$$\boldsymbol{y = b_0 + \sum_{k=1,m} b_k x_k}$$

ISSN 2395-8618

Then we compute b_k by minimizing:

$$\sum_{i=1,n}(y_i - b_0 - \sum_{k=1,m} b_k x_{ki})^2$$

Thus, it determines a hyper-plane which is a least square approximation of data points. If data is mean-centered, then b_0=0. It is advised to mean-center data to simplify the computations.

It is interesting to note that as a result of linear regression, the data points *may not* be at least distance from the regression line. Here we present an algorithm using PCA that results in a better least distance line. There are two ways in which regression analysis is improved: data reduction and hybrid algorithm. As a first step, PCA is used on the dataset for data reduction. For improved performance in the second step, we create a hybrid linear regression algorithm coupled with PCA.

IMPROVED LINEAR REGRESSION

Input: array of data points (*x, y*)
Output: line *y=a + bx*
Method:
Traditional: compute *a* and *b*, by minimizing

$$\sum_{i=1,n}(y_i - a - bx_i)^2$$

Let error1 be the computed traditional error value.
New: compute *a* and *b*, by minimizing

$$\sum_{i=1,n}\frac{(y_i - a - bx_i)^2}{\sqrt{(1+b^2)}}$$

Let error2 be the computed PCA adapted error value.
Compare error1 and error2

Example: We have a dataset of randomly created 20 points. Matlab computes the regression line as the red line, see Figure 4. PCA computes the blue line. As can be visually seen, the blue line is the least distance line instead of the red regression line. For direction vectors and approximation error of data points from the line, see Table 4.

Both linear regression and PCA compute vectors so that the variation of observed points from the computed vector is minimum. However, as shown in Figure 4, they give two different vectors, red via linear regression, and blue via PCA. The approximation error indicates the PCA adapted regression line is a better approach than by LSA method.

Data reduction is attributed to the non-zero eigenvalues of the data matrix *A*. Since $m \times n$ data matrix is decomposed into $A = USV^T$ where *U* is $m \times m$, *S* is $m \times n$, and *V* is $n \times n$. If there are only *k* non-zero eigenvalues where $k \leq \min(m, n)$, the matrix *A* has lossless representation by using only *k*

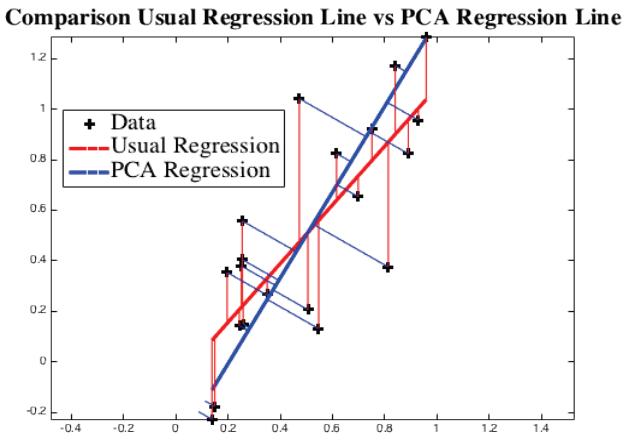


Fig. 4: Using the data points "+", traditional linear regression line is shown in red (LSA) and a principal component is shown in blue (PCA). Visually we can see that the points are much closer to blue line than to the red line.

TABLE IV
COMPARISON OF LINEAR REGRESSION METHODS

For usual regression line	
Direction vector	[0.642388, 0.766380]
Relative Regression Error	0.391169
For PCA adapted regression line	
Direction vector	[0.514757,0.857336]
Relative Regression Error	0.173438

columns of *U*, *k* columns of *V* and *kxk* diagonal matrix *S*. If an eigenvalue is very small as compared to others, then ignoring it can lead to further data reduction while retaining most of the information.

B. PCA and Logistic Regression

Along with linear regression, logistic regression (log linear) has been a popular data mining technique. However, both when used stand alone, have limited ability to deal with inter dependent factors. Linear regression is suitable for data that exhibits linear relation, but as all data does not have linear trend, the Logistic models estimate the probability and is applicable to "S-shaped" data. This model is particularly suitable for population growth with limiting condition. As it was with linear regression, it is beneficial to use logistic regression when coupled with PCA.

Population growth is described by exponential function; population is controlled by the limiting condition. The liver disease model is a composition of these two functions as shown below. The mapping from linear to logistic function is described as follows [23].

Thus for logistic function P(x)∈(0, 1) instead of linear function *P(x) = a + bx*, the function becomes:

$$P(x) = \frac{e^{a+bx}}{1+e^{a+bx}}$$

ISSN 2395-8618

To solve for a and b, we write:

$$log\frac{P(x)}{1-P(x)} = a + bx$$

We make use of PCA in designing better logistic regression algorithm, presented below. Here the hybrid algorithm is presented for two-dimensional data, however, it can be easily extended to higher dimensions.

IMPROVED NON-LINEAR LOGISTIC REGRESSION

Input: array of data points (x, y)
Output: non-linear PCA adapted logistic function
Method: For logistic regression, map

$$y \rightarrow log_e(\frac{y}{1-y})$$

Apply *improved* regression line to y values computed from (*new approach*) line $y=a + bx$
Map y values back

$$y \rightarrow \frac{e^y}{1+e^y}$$

Example: In this example, we have a training dataset of 20 students obtained from [24] who studied for the exam for given hours (horizontal axis) and passed or failed (vertical axis) the test. The curves are the trained logistic regression predictor for chances of passing the exam. Fail and pass are coded numerically as 0 and 1, see Figure 5.

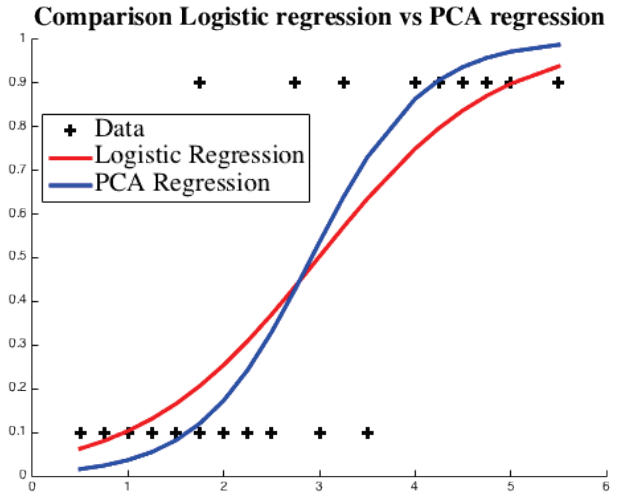


Fig. 5: Using the data points "+", usual logistic regression curve is given in red and the regression curve generated by the proposed hybrid model is given in blue.

The approximation errors for are shown in Table 5. The example exhibits that the PCA (blue) curve is a better approximation predictor with 60% less approximation error.

TABLE V
COMPARISON OF LOGISTIC REGRESSION METHODS

Logistic Regression Relative Error	0.443061
PCA Regression Relative Error	0.157216

C. *How do we measure the goodness of a model?*

In data mining there are standard measures, called gold standard, for labeling and measuring the prediction accuracy. These measures are useful in comparing the results of classification. Here we will list three metrics, Precision (P), Recall (R) and F-1. In these metrics, actual value and predicted value are used to create a label for each instance outcome, where the instance outcome is either positive or negative. The labels used here are True Positive (TP), False Positive (FP) and False Negative (FN) to measure the accuracy (the reader is encouraged to consult [17] for further details). After labeling, we define the Precision (P), Recall (R) and F-metrics to measure the effectiveness of the prediction on (1) correct prediction on all positive instances and (2) TP prediction on a sample of instances (positive and negative) under investigation on the training data.

$$Precision(P) = \frac{TP}{TP+FP}$$

$$Recall(R) = \frac{TP}{TP+FN}$$

The measure F-1 is the weighted Harmonic average of P and R. It is the reciprocal of the weighted average of the reciprocals of P and R.

For $0 \leq \alpha \leq 1$, it simplifies to:

$$F_1 = \frac{PR}{\alpha R + (1-\alpha)P}$$

For $\alpha = 0$, it turns out to be the *recall* measure R, for $\alpha = 1$, it becomes *precision* measure P, and for $\alpha = 1/2$, it further simplifies to *traditional* measure

$$F_1 = \frac{2PR}{P+R}$$

$$or\ F_1 = \frac{2TP}{2TP+FP+FN}$$

This is the preferred measure when there are fewer misses of both positive and negative instances, (i.e. both FN and FP are small) [17].

Goodness of fit is an interesting analysis criterion. Our goal is to show how the hybrid linear/logistic regression model is better than the more straightforward measures generated for linear/logistic measures.

V. DIMENSIONALITY REDUCTION USING PCA

The nature of data dictates how many dimensions can be reduced. The data is not just the attributes, but the dependency

ISSN 2395-8618

TABLE VI
NUMERIC VALUES FOR 9-INTERSECTIONS REQUIRED TO DISTINGUISH EACH RCC8 RELATION

	IntInt	BndBnd	BndInt	IntBnd	IntExt	BndExt	ExtInt	ExtBnd	ExtExt	RCC8
ObjectPair1	1	1	1	1	0	0	0	0	**0**	DC
ObjectPair2	1	0	1	1	0	0	0	0	**0**	EC
ObjectPair3	0	0	1	1	1	1	1	1	**0**	EQ
ObjectPair4	0	1	1	0	0	0	1	1	**0**	NTPPc
ObjectPair5	0	0	1	0	0	0	1	1	**0**	TPPc
ObjectPair6	0	1	0	1	1	1	0	0	**0**	NTPP
ObjectPair7	0	0	0	1	1	1	0	0	**0**	TPP
ObjectPair8	0	0	0	0	0	0	0	0	**0**	PO

and redundancy among the attributes. If $A=USV^T$ is full dimension SVD of A, where A is $m \times n$, U is $m \times m$, V is $n \times n$, S is $m \times n$, then the total size for decomposition representation of A is $m \times m + n \times n + m \times n$, which is larger than $m \times n$, the size of A. Our goal is to find an integer k smaller than m and n, and use first k columns of U and first k columns of V and restrict S to first k eigenvalues to show the effect of dimensionality reduction. Since eigenpairs are sorted on descending order of variance, deleting the least variation components do not cause significant error in data [25].

In practice, it is not our intent to reconstruct the original matrix A from reduced USV^T but to view the data from a fresh perspective and to use the reduced representation to extract information hidden in the original representation. PCA is used to reduce dimensionality in the new space, not the original space. Our purpose to see if we can leverage PCA to reduce dimensionality in the original space. This is an open question.

ALGORITHM FOR DATA DIMENSIONALITY REDUCTION

Input: $m \times n$ data matrix A

Output: reduced data $m \times k$ matrix B

Steps

Create covariance matrix $C = A^TA$

Compute the eigenvalues and eigenvectors of C

Rank the eigenvectors on descending order of eigenvalues: U, S, V

Normalize the columns to unity

Make diagonal entries of U, V as non-negative

Choose k using one of the criteria described above, k less than or equal to the number of non-zero eigenvalues.

Construct the transform matrix $V_{n \times k}$ from the selected k eigenvectors.

Transform A to $AV_{n \times k}$ in eigenspace to express data in terms of set of eigenvectors reduced from n to k.

It gives a new set of basis vectors and a reduced k-dimensional subspace of k vectors where the data resides.

A. Dimension Reduction in Qualitative Spatial Reasoning

PCA has been used mainly with numerical data. If the data is categorical or logical, then data is first converted to numerical. We will see how PCA has the ability to resolve and isolate spatial-temporal patterns in the data presented in Table 2. We present a new robust PCA enabled method for QSR. Table 2 describes eight topological relations between pairs of spatial objects. The values of entries are true and false. In order to use PCA, we first covert the logical data to numerical data. We use 1 for false and 0 for true, see Table 6. Our goal is row dimension reduction. As intersection is a complex operation and also computationally expensive, we want to reduce the number of intersections required. For example, for 1000 pairs of objects, there will be 9000 pairwise intersections. By eliminating one intersection, we can reduce 9000 to 8000 intersections, almost 11% improvement in execution. We will show that PCA gives insights, using which we can do better. In fact, we are able to reduce 9000 to 4000 intersections. This is more than 55% reduction in computation time! This means *we can replace 9-Intersection model by 4-Intersection model* which is now applicable to spatial as well as non-spatial objects, like web documents.

For RCC8, the item-attribute-concept becomes object pair--9-Intersection—relation classification. In Table 6, row header represents a pair of spatial objects, column headers are the 9-Intersection attributes, and last column RCC8 is the classification of the relation based on the intersections. Table 2 and Table 6 show a sample of eight pairs of objects, one of each classification type.

We consider Table 6 is an 8×9 input matrix A. On using Matlab SVD on A^TA, we get nine eigenvectors and nine eigenvalues of A^TA shown in Table 7. Since five eigenvalues are zero, the corresponding eigenvectors are useless. This tells us that nx9 data can be replaced with nx4 right away without any loss of information.

In Table 8, first row enumerates the eigenvalues of A^TA. The next rows represent the error on using first k eigenpairs (where k is the column number). *newA* is USV^T based on k eigenpairs used. Error Original is the $|A\text{-}newA|/|A|$ percentage error in the original space. Error Projection is the $|AV\text{-}newAV|/|AV|$ percentage error in the projection space. Error Eigenvalue is the $\sum_{p=k+1,9} \lambda_p / \sum_{p=1,9} \lambda_p$ percentage error in the

ISSN 2395-8618

TABLE VII
EIGENVECTORS FOR RCC8, ROWS ARE EIGENVECTORS; LAST COLUMN IS EIGENVALUES

	Eigenvectors										Eigenvalues
V1 =	[0.2833	0.4914	0.4914	0.1723	0.319	0.319	0.319	0.319	0	]	3.8793
V2 =	[0	0.4082	–0.4082	0	–0.4082	–0.4082	0.4082	0.4082	0	]	2.4495
V3 =	[0.249	0.2593	0.2593	0.5946	–0.3353	–0.3353	–0.3353	–0.3353	0	]	2.0405
V4 =	[–0.926	0.2202	0.2202	0.2129	0.0073	0.0073	0.0073	0.0073	0	]	1.337
V5 =	[0	0.3108	–0.2504	–0.0605	0.3027	–0.0524	0.4338	–0.7446	0	]	0
V6 =	[0	0.2159	0.4498	–0.6657	–0.5075	0.0576	0.001	–0.2169	0	]	0
V7 =	[0	0	0	0	0	0	0	0	1	]	0
V8 =	[0	0.1917	0.1443	–0.336	0.5191	–0.6634	–0.3211	0.1294	0	]	0
V9 =	[0	–0.5442	0.4363	0.108	–0.0205	–0.4157	0.5725	–0.0282	0	]	0

TABLE VIII
EIGENVALUES, EIGENVECTORS, AND PERCENT ERRORS FOR RCC8 REDUCTION USING PCA

	Eigenvalues								
Eigenvalues	3.879290	2.44949	2.040497	1.336968	0	0	0	0	0
Error Original	66.53072	46.948	25.72995	0	0	0	0	0	0
Error Eigenvalue	66.53072	46.948	25.72995	0	0	0	0	0	0
Error Quantization	100	74.535599	27.216553	0	0	0	0	0	0
Error Projection	60.033047	34.79682	13.7743	0.000001	0	0	0	0	0

eigenvalue space. Error Quantization uses quantization before error calculation. For example, now with quantization using 4 eigenvalues we see that there is no error between *newA* and *A*. This is what we expected as *A* has boolean elements.

Table 8 shows that the transformed space created using 4 eigenvalues retains perfect information. This means that for all object pairs, the relations can be described with 4 eigenvectors as an $n \times 4$ matrix instead of nx9 matrix with zero error. So how can we deduce the original dimensions that retain most of the information? We explore that in the next section.

VI. EXPERIMENTS AND OUTCOMES

Here we show the application of PCA for dimension reduction in qualitative spatial reasoning and liver disease data. In addition decision tree is used for spatial data classification whereas the improved logistic regression is applied to liver disease data classification.

A. Qualitative Spatial Reasoning

In Section IV, we determined that 4 attributes are sufficient to classify QSR relations in the transformed space. However, it does not tell anything about the attributes in the original space. Now we will see if we can translate this new found knowledge into the original space of Table 2. How can we find four intersection attributes that will lead to the 8 distinct topological relations?

From careful observation of Table 2 we see that the IntInt and BndBnd columns have the most useful information in the sense that they are sufficient to partition the RCC8 relations into eight jointly exhaustive and pairwise distinct (JEPD) classes, which can be further grouped into three classes: {DC, EC}, {NTTP, NTTPc}, and {EQ, TPP, TPPc, PO}.

We revisit Table 2 as Table 9 by shading some entries and analyze them. It shows that only 4-intersections are sufficient for classification of topological relations. The nature of data suggests that the remaining attributes are not necessary. This table can be interpreted and formulated in terms of rules for system integration. These rules are shaded and displayed for visualization in the form of Table 9.

TABLE IX
RCC8 RELATIONS ATTRIBUTES FOR CLASSIFICATION,
(REVISITING TABLE 2)

	Int Int	***Bnd Bnd***	***Int Bnd***	***Bnd Int***	Int Ext	Bnd Ext	Ext Int	Ext Bnd	Ext Ext
DC	**F**	**F**	F	F	T	T	T	T	T
EC	**F**	**T**	F	F	T	T	T	T	T
NTPP	**T**	**F**	**F**	T	F	F	T	T	T
NTPPc	**T**	**F**	**T**	F	T	T	F	F	T
EQ	**T**	**T**	**F**	**F**	F	F	F	F	T
TPP	**T**	**T**	**F**	**T**	F	F	T	T	T
TPPc	**T**	**T**	**T**	**F**	T	T	F	F	T
PO	**T**	**T**	**T**	**T**	T	T	T	T	T

Thus, Table 9 reveals that the spatial relations can be specified by at most four intersection attributes. The shaded columns of Table 9 are transcribed into a decision tree for easy visualization of the rules to classify the RCC8 eight relations, see Figure 6.

This conclusion makes no assumptions about the objects being spatial or non-spatial as long as they are valid. In addition, this analysis is applicable to discrete and continuous objects alike.

ISSN 2395-8618

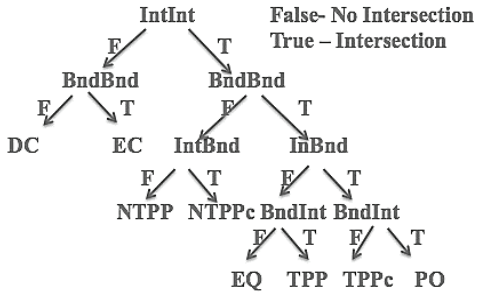


Fig. 6. Classification tree for the topological relations, where T and F represent whether the objects intersect or not respectively.

TABLE X
EIGENPAIRS AND ERROR IN DATA REDUCTION.

	Raw Data	Normalized Data
Eigenvalues	−8635.3	38.6315
Eigenvectors	$\begin{Bmatrix} 0.0875 \\ 0.0115 \\ 0.0196 \\ 0.0125 \\ 0.4838 \\ 0.4376 \\ 0.7523 \\ 0.0107 \\ 0.0032 \\ 0.0015 \end{Bmatrix}$	$\begin{Bmatrix} 0.3666 \\ 0.2201 \\ 0.3439 \\ -0.0374 \\ -0.4160 \\ -0.4139 \\ -0.4026 \\ 0.3966 \\ -0.0552 \\ 0.1753 \end{Bmatrix}$
Errors	0.367508	0.671054

B. Health Informatics

We will use public domain dataset from UCI Machine Learning Repository [22] to automate the simplicity, applicability and usability of our approach.

For application of our algorithm, we selected liver disease classification dataset. This dataset was selected particularly as most of its attribute were numeric and classification attribute is binary representing presence or absence of the disease. This dataset is compatible with logistic regression and is also well suited to PCA that processes numerical values only.

We obtained the dataset from Machine Learning Repository at the University of California, Irvine [22]. The dataset contains liver disease information about 583 patients out of which 416 are with liver disease and 167 are healthy. The dataset consists of 441 male and 142 female patients. The liver disease classification is based on 10 parameters: age of the patient, gender of the patient, total Bilirubin, direct Bilirubin, Alkaline Phosphotase, Alamine Aminotransferase, Aspartate Aminotransferase, total Proteins, Albumin, Albumin and Globulin ratio. There were two types of recommendations based on these experiments: patient has liver disease or patient does not have liver disease.

This is a fairly small size dataset for classification of 583 patients. Learning from this dataset can be used to predict possible disease for a new patient quickly without further analysis. The goal is not data mining per se, but to show the feasibility of improved algorithms over the existing algorithms and data reduction to classify liver disease. The reduction in one attribute reduces the data size by 9%. We applied PCA on the data to reduce 10 attributes to 3 or 4 attributes, which contribute the most to the eigenvector corresponding to the highest eigenvalue, while retaining approximately the same predictive power as the original data.

For experiment, we created two versions of the dataset: first dataset is raw, the second dataset is mean-centered with unit standard deviation. PCA determines that there is only one non-zero eigenvalue all other eigenvalues are insignificant. One non-zero eigenvalue is shown in Table 10.

This indicates that only a single attribute in the transformed space is sufficient to diagnose the patients. But this does not tell us which original attributes contributed to reduction. The principal components on normalized data are more realistic in this case, as the normalized data attributes values are evenly distributed. For nominal attributes, mapping nominal to numerical can make a difference. However, covariance and correlation approaches are complementary.

The principal component corresponding to non-zero eigenvalue is a linear combination of original attributes. Each coefficient in it is a contribution of the original data attributes. How do we select the fractions of original attributes because the coefficients in this vector are real?

The only thing it means is that each coefficient is a fractional contribution of the original data attributes. It is clear that the three (Alkaline Phosphatase Alamine Aminotransferase, Total Proteins, Aspartate Aminotransferase) of the ten coefficients are more dominant than the others, however normalized data analysis found three more slightly less dominant coefficients. In either case, the contribution of the original three attributes is more than 95%. Eliminating the other attributes, we compute the approximation error due to dimension reduction to three attributes.

PCA analysis shows that even after 60% reduction, using only 40% of data, the precision is almost the same whereas the gain in computation performance is significant, see Figure 7. For recall, the reduced data regression misses more negatives see Table 11. It is preferable to miss less positive than more negatives. Table 11 corresponds to traditional logistic regression Table 12 corresponds to hybrid logistic regression algorithm. It shows that hybrid algorithm consistently outperforms the traditional algorithms.

VII. CONCLUSION

Principal components analysis is a procedure for identifying a smaller number of uncorrelated variables, called "principal components", from a large set of data. The goal of principal components analysis is to explain the maximum amount of variance with the fewest number of principal components.

ISSN 2395-8618

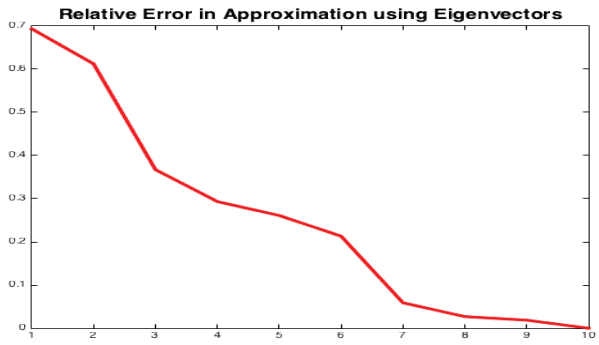


Fig. 7. Error in estimating the original data from eigenvectors where x-axis represents the number of eigenvectors used in data approximation (starting from the most significant to the least significant one) and y-axis represents the error percentage of the estimation.

Principal components analysis is commonly used as one step in a series of analyses. We use principal components analysis to reduce the number of variables and avoid multi-collinearity, or when we have too many predictors relative to the number of observations. We have used PCA in two diverse genres, QSR and Health Informatics to improve traditional data reduction and regression algorithms.

QSR uses 9-Intersection model to determine topological relations between spatial objects. In general, PCA utilizes numerical data for analysis and as QSR data is logical bivalent, we mapped the logical data to numerical data. PCA determined that 4-attributes are adequate in the transformed space. In general, reduction in transformed space does not tell anything about reduction in base space. However, in this case study, we leveraged PCA to determine the possibility of reduction in the base space. We succeeded in achieving similar reduction the original space of RCC8 relations. This yields more than 55% efficiency in execution time.

We also presented hybrid algorithms that adaptively used PCA to improve the linear and logistic regression algorithms. With experiments, we have shown the effectiveness of the enhancements. All data mining applications that dwell on these two algorithms will benefit extensively from our enhanced algorithms, as they are more realistic than the traditional algorithms. The tables in the paper body vouch for this improvement. We applied our algorithms to the Liver Patient dataset to demonstrate the usability and applicability of our approach, especially in the area of health related data.

TABLE XI
ERROR COMPARISON METRICS *TRADITIONAL* ALGORITHM

	Raw	PCA 40% Data
Precision	0.715	0.743
Recall	0.995	0.940

TABLE XII
ERROR COMPARISON METRICS *HYBRID* ALGORITHM

	Raw	PCA 40% Data
Precision	0.737	0.757
Recall	0.987	0.937

VIII. REFERENCES

[1] M. J. Egenhofer, R. Franzosa, "Point-Set topological Relations", *International Journal of Geographical Information Systems* 5(2), pp.161–174, 1991.

[2] D.A. Randell, Z. Cui, A.G. Cohn, "A Spatial Logic Based on Regions and Connection". KR92:165–176, 1992.

[3] C. Sabharwal, J. Leopold, and Nathan Eloe, "A More Expressive 3D Region Connection Calculus", Proceedings of the 2011 International Workshop on Visual Languages and Computing DMS'11, Florence, Italy, Aug. 18–20, 2011, pp. 307–311, 2011.

[4] *The Promise of Big Data*, Harward School of Public Health Magazine, pp. 15–43, 2012

[5] A. Bernard: *Healthcare Industry Sees Big Data As More Than a Bandage*, CIO, 2013.

[6] C. M. Bishop, *Pattern Recognition and Machine Learning*, 2006 Springer

[7] *The Global Use of Medicines: Outlook Through 2016*, IMS Institute for Healthcare Informatics, pp.1–36, 2012

[8] S. Israel: *Contextual Health vs The Elephant in the Hospital*, Forbes, Tech, pp.1–10, 2013

[9] J. Bort: *How the CDC Is Using Big Data to Save You from the Flu*, Business Insider

[10] A. Parmar, "Want to know if you will develop high blood pressure next year? With big data magic you can", *MedCity News,* 2012

[11] P. Groves, B. Kayyali, D. Knott, S. V. Kuiken, "The 'Big Data' Revolution in Healthcare", *Center of US Health System Reform Business Technology Office,* pp. 1–20, 2013

[12] B.W. Chee, R. Berlin, B. Schatz, "Predicting Adverse Drug Events from Personal Health Messages". *In: Annual Symposium Proceedings*, pp. 217–226 (2011).

[13] X. Liu, and H. Chen, 2013. "Azdrugminer: An Information Extraction System for Mining Patient Reported Adverse Drug Events in Online Patient Forums," *Smart Health. Springer*, pp. 134–150.

[14] C. C. Yang, L. Jiang, H. Yang, M. Zhang, "Social Media Mining for Drug Safety Signal Detection" *ACM SHB'12*, October 29, 2012, Maui, Hawaii, USA.

[15] H. Yang and C. C. Yang, "Using Health-Consumer-Contributed Data to Detect Adverse Drug Reactions by Association Mining with Temporal Analysis", *ACMTrans. Intell. Syst. Technol*. 6, 4, Article 55 (July 2015)

[16] H. Hotelling, "Analysis of a complex of statistical variables into principal components" *Journal of Educational Psychology*, 24, 417–441, and 498–520, 1993

[17] H. Jim: Linear Algebra 2014 http://joshua.smcvt.edu/linearalgebra

[18] J. Shlens *A Tutorial on Principal Component Analysis*, arXiv:1404.1100 [cs.LG], pp. 1–15,2014.

[19] K. Baker, Singular Value Decomposition Tutorial, www.ling.ohio-state.edu/~kbaker/pubs/Singular_Value_Decomposition_Tutorial.pdf, January 2013

[20] H. Y. Chen, R. Liégeois, J. R. de Bruyn, and A. Soddu, "Principal Component Analysis of Particle Motion", *Phys. Rev.* E 91, 042308, 15 April 2015

[21] Ling Jiang, Christopher C. Yang, and Jiexun Li, "Discovering Consumer Health Expressionsfrom Consumer-Contributed Content" SBP 2013, LNCS 7812, pp. 164–174, 2013

[22] UCI Machine Learning Repository, Liver Patient Dataset https://archive.ics.uci.edu/ml/datasets/

[23] A. Goel, R. G Pinckney and B. Littenberg, APACHE II Predicts Long-term Survival in COPD Patients Admitted to a General Medical Ward, *J Gen Intern Med.* 2003 Oct; 18(10): 824–830.

[24] Wikipedia: https://en.wikipedia.org/wiki/F1_score

[25] J. Leskovec, A. Rajaraman, J. D Ullman, Datamining of Massive Datasets, 2014

ISSN 2395-8618

A Segment-based Weighting Technique for URL-based Genre Classification of Web Pages

Chaker Jebari

***Abstract*—We propose a segment-based weighting technique for genre classification of web pages. This technique exploits character n-grams extracted from the URL of the web page rather than its textual content. The main idea of our technique is to segment the URL and assigns a weight for each segment. Experiments conducted on three known genre datasets show that our method achieves encouraging results.**

***Index Terms*—URL, genre classification, web page, segment weight.**

I. Introduction

As the World Wide Web continues to grow exponentially, the classification of web pages becomes increasingly important in web searching. Web page classification, assigns a web page to one or more predefined classes. According to the type of the class, the classification can be divided into sub-problems: topic classification, sentiment classification, genre classification, and so on.

Currently, search engines use keywords to classify web pages. Returned web pages are ranked and displayed to the user, who is often not satisfied with the result. For example, searching for the keyword "*Java*" will provide a list of web pages containing the word "*Java*" and belonging to different genres such as "*tutorial*", "*exam*", "*Call for papers*", etc. Therefore, web page genre classification could be used to improve the retrieval quality of search engines [18]. For instance, a classifier could be trained on existing web directories and be applied to new pages. At query time, the user could be asked to specify one or more desired genres so that the search engine would returns a list of genres under which the web pages would fall.

However, although potentially useful, the concept of "*genre*" is difficult to define and genre definitions abound. Generally speaking, a genre is a category of artistic, musical, or literary composition characterized by a particular style, form, or content, but more specialized characterizations have been proposed [23]. For instance, [14] defined a genre as a bundle of facets, focusing on different textual properties such as brow, narrative, and genre. According to [25], the genres found in web pages (also called cyber-genres) are characterized by the triple <content, form, functionality>. The content and form attributes are common to non-digital genres and refers to the text and the layout of the web page respectively. The functionality attribute concerns exclusively digital genres and describes the interaction between the user and the web page.

A web page is a complex object that is composed of different sections belonging to different genres. For example, a conference web page contain information on the conference, topics covered, important dates, contact information and a list of hypertext links to related information. This complex structure need to be captured by a multi-label classification scheme in which a web page can be assigned to multiple genres [23, 28, 9].

A broad number of studies on genre classification of web documents have been proposed in the literature [23]. These studies differ mainly with respect to the feature set they use to represent web documents. These features are divided into four types: surface features (function words, genre specific words, punctuation marks, document length, etc), structural features (Parts Of Speech (POS), Tense of verbs, etc), presentation features (number of particular HTML tags and links) and contextual features (URL, keywords, etc.).

Once a set of features has been obtained, it is necessary to choose a classification algorithm. These are often based on machine learning techniques such as Naïve Bayes, K-Nearest Neighbour, Decision trees, Support Vector Machine, Neural Networks, and Centroid-based techniques [20].

It is worth noting that many researchers study the usefulness of URL for genre classification of web documents [21, 22, 28] without deeply exploiting its content and the structure. With respect to this, we proposed in this paper a new approach that represents a web page by a bag of character n-grams (contiguous n-characters) extracted only from the URL. Using only the URL, we eliminate the necessity of downloading the web page. It is very useful when the web page content is not available or need more time/space to display. Moreover, we proposed in this paper a new weighting technique that exploits the URL structure.

The remainder of the paper is organized as follows. Section 2 present previous works on genre classification of web pages. Section 3 describes the extraction of character n-grams from the URL. A new segment-oriented weighting technique is also presented at the end of Section 3. The evaluation of our approach is described in Section 4. Finally, Section 5 concludes our paper with future research directions.

Manuscript received on October 9, 2015, accepted for publication on January 30, 2016, published on June 25, 2016.

Chaker Jebari is with College of Applied Sciences, Ibri, Sultanate of Oman (e-mail: jebarichaker@yahoo.fr).

ISSN 2395-8618

II. Previous Work on Genre Classification of Web Pages

The previous works on genre classification of web pages differ with respect to the following three factors:

(1) The list of genres used in the evaluation, called also genre palette
(2) The features used to represent the web page
(3) The classification method used to identify the genre of a given web page

The last factor is often based on machine learning techniques such as Naïve Bayes, *K*-Nearest Neighbour, Decision trees, Support Vector Machine, Neural Networks, and Centroid-based techniques. These machine-learning techniques were deeply studied by Mitchell [20]; therefore, we will focus more on the first two factors.

A genre palette can be general if it covers a large part of web or specific if it aims to study a limit number of genres. To study the usefulness of web genres, Meyer and Stein [19] compiled the KI-04 corpus which is composed of 1205 web pages distributed over 8 genres (article, download, link collection, private portrayal, non-private portrayal, discussion, help and shop) (See Table 4). Based on a corpus of 15 genres, Lim et al. [16] investigated the usefulness of information found in different parts of the web page such (title, body, anchor, etc.). Kennedy and Shepherd [13] proposed a more specific and hierarchal genre palette that contains two super-genres (home pages, non-home pages). The super-genre home page is divided into two three sub-genres (personal, corporate and organization pages). In her master thesis, Boese [3] collected a corpus of 343 web documents distributed across 10 structured genres (abstract, call for papers, FAQ, How-to, Hub/sitemap, Job description, Resume/CV, Statistics, Syllabus, Technical paper). Santini [23] compiled manually a corpus of 1400 web pages equally distributed across seven genres (blogs, eshops, FAQs, front pages, listings, personal home pages, search pages). To examine the effects of web evolution on the task of classifying web pages by genre, Boese and Howe [4] used the dataset WebKB [5], which contains 8282 web pages. In this dataset, 4518 web pages belong to one of six functional genres (course, department, faculty homepage, project, staff homepage, and student homepage). The remaining 3764 web pages are assigned to the genre "*Other*." To evaluate a multi-label genre classification schema, Vidulin et al. [27] used the web site Google Zeitgeist to build the multi-label corpus 20-genre. This corpus contains 1539 web pages belonging to 20 genres (See Table 5). More recently, Priyatam et al. [22] investigated the usefulness of URL to classify web pages into two categories (health and tourism). For this reason, they collected and tagged manually 3000 web pages.

Many types of features have been proposed for automatic genre categorization. These features can be grouped on four groups. The first group refers to surface features, such as function words, genre specific words, punctuation marks, document length, etc. The second group concerns structural features, such as Parts Of Speech (POS), Tense of verbs, etc. The third group is presentation features, which mainly describe the layout of document. Most of these features concerns HTML documents and cannot be extracted from plain text documents. Among these features, we quote the number of specific HTML tags and links. The last group of features is often extracted from metadata elements (URL, description, keywords, etc.) and concerns only structured documents.

With respect to plain text document representation, Kessler et al. [14] have used four types of features to classify the Brown corpus by genre. The first types are structural features, which includes counts of functional words, sentences, etc. The second types are lexical features, which includes the existence of specific words or symbols. The third kinds of features are character level features, such as punctuation marks. The last kind concerns derivative features, which are derived from character level and lexical features. These four features sets can be grouped on two sets, structural features and surface features. Karlgren [12] have used twenty features: count of functional words, POS count, textual count (e.g. the count of characters, the count of words, number of words per sentence, etc.), and count of images and links. Stamatatos et al. [24] identified genre based on the most English common words. They have used the fifty most frequent words on the BNC corpus[2] and the eight frequent punctuation marks (period, comma, colon, semicolon, quotes, parenthesis, question mark, and hyphen). Dewdney et al. [6] have adopted two features sets: BOW (Bag of Words) and presentation features. They used a total of 89 features including layout features, linguistic features, verb tenses, etc. Finn and Kushmerick [7] used a total of 152 features to differentiate between subjective vs. objective news articles and positive vs. negative movie reviews. Most of these features were the frequency of genre-specific words.

With respect to web page representation, Meyer and Stein [19] used different kinds of features including presentation features (i.e. HTML tag frequencies), classes of words (names, dates, etc.), and frequencies of punctuation marks and POS tags. Lim et al. [16] introduced new sets of features specific to web documents, which are extracted from URL and HTML tags such as title, anchors, etc. First, Kennedy and Shepherd [13] used three sets features to discriminate between home pages from non-home pages. Secondly, they classify home pages into three categories (personal, corporate, and organization). Their feature set comprises features about the content (e.g., common words, Meta tags), form (e.g., number of images), and functionality (e.g., number of links, use of JavaScript).

Vidulin et al. [27] used 2,491 features divided into four

ISSN 2395-8618

groups: surface, structural, presentation and context features. Surface features include function words, genre-specific words, sentence length and so on. Structural features include Part Of Speech tags, sentence types and so on. Presentation features describe the formatting of a document through the HTML tags, while context features describe the context in which a web page was found (e.g. URL, hyperlinks, etc.).

Kim and Ross [15] used image, style, and textual features to classify PDF documents by genre. The image features were extracted from the visual layout of the first page of the PDF document. The style features are represented by a set of genre-prolific words, while textual features are represented by a bag of words extracted from the content of the PDF document. Kim and Ross pointed out that some PDF are textually inaccessible due to password protection, and that image features would be especially useful in this case.

In his PhD thesis, Jebari [8] exploits the features extracted from three different sources, which are the URL addresses, the title tag, the heading tags, and the hypertext links. The experiments conducted on the two known corpora KI-04 and WebKB show that combining all features gave better results than using each feature separately. Kanaris and Stamatatos [11] used character n-grams and HTML tags to identify the genre of web pages. They stated that character n-grams are language-independent and easily extracted while they can be adapted to the properties of the still evolving web genres and the noisy environment of the web. In her thesis study, Mason [17] used character n-grams extracted from the textual content to identify the genre of a web page. Recently, Myriam and David [21] proposed a new genre classification of web pages that is purely based on URL. Their approach is based on the combination of different character n-grams of different lengths. More recently, Priyatam et al. [22] used character *n*-grams extracted from the URL to classify Indian web pages into two genres: sport and health. They aim to improve the Indian search engine Sandhan.

III. Web page representation

The representation of a web page is the main step in automatic genre classification. The first paragraph of this section describes the extraction of features from the URL and the second paragraph presents a new Weighting technique that exploits the URL segments.

A. Feature extraction

Often, features for classifying web pages are extracted from its content, which needs more time since it requires downloading it previously [1]. To deal with this issue, we decided in this paper to represent a web page by its URL, since every web page possesses a URL, which is a relatively small string (therefore easy to handle).

A URL can be divided into the following segments: Domain Name, Document Path, Document Name, and Query string [2]. For example for the URL: *http://www.math.rwth-aachen.de/~Greg.Gamble/cv.pdf*, we can extract the following segments:

- Domain name (DOMN): www.math.rwth-aachen.de
- Document path (DOCP) : ~Greg.Gamble
- Document name and query string (DOCN): cv.pdf

For each URL segment we performed some pre-processing, which consist into

- Removing special characters (_, . , : , ? , $, %) and digits.
- Removing common words (for example the word "*www*" from the domain name and the words "*pdf*", "*html*", etc. from the document name)
- Removing generic top-level domains (.edu, .uk, .org, .com, etc.) from the domain name
- Removing words with one character.

After that, we extracted from each word all character n-grams. For example, from the word "*JAVA*" we can extract one character 4grams (JAVA), two character 3-grams (JAV, AVA) and 3 character 2-grams (*JA*, *AV*, *VA*).

To reduce the time needed for training and testing, we removed the words and character *n*-grams that appear in less than 10 web page URLs.

B. Segment-based Weighting Technique

Term Frequency does not exploit the structural information present in the URL. For exploiting URL structure, we must consider not only the number of occurrences of character *n*-gram in the URL but also the URL segment the character *n*-grams are present in.

The idea of the proposed weighting technique is to assign greater importance to character *n*-grams that belong to the URL segment, which is more suitable to represent a web page. To implement this idea, we proposed a new weighting technique termed SWT. In this technique, the weight for a given character *n*-gram C_i in a URL U_j is defined as follows:

$$SWT(C_i,U_j)=\sum_s W(s)\cdot TF(C_i,s,U_j) \tag{1}$$

where

- $TF(C_i,s,U_j)$ denotes the number of times the character *n*-gram C_i occurs in the segment s of the URL U_j
- $W(s)$ is the weight assigned to the segment s and is defined as follows:

$$W(s)=\begin{cases}\alpha & if \quad s=DOMN\\ \beta & if \quad s=DOCP\\ \lambda & if \quad s=DOCN\end{cases} \tag{2}$$

Where the values of the weighting parameters α, β and λ are determined using an experimental study.

ISSN 2395-8618

IV. Evaluation

A. Datasets

To evaluate our approach, we used three datasets: KI-04, 20-Genre and KRYS-I. These datasets are unbalanced, which means that the web documents are not equally distributed among genres.

1) KI-04

This dataset was built following a palette of eight genres suggested by a user study on genre usefulness. It includes 1295 web pages, but only 800 web pages (100 per genre) were used in the experiment described in Meyer and Stein [19]. In the experiments described in this paper, I have used 1205 web pages because we have excluded empty web pages and error messages; see Table I.

Table I
Composition of KI-04 dataset

Genre	# of web pages
Article	127
Download	151
Link collection	205
Private portrayal	126
Non-private portrayal	163
Discussion	127
Help	139
Shop	167

2) 20-Genre

This dataset, gathered from internet, consists of 1539 English web pages classified into 20 genres as shown in the following table. In this dataset, each web page was assigned by labelers to primary, secondary, and final genres. Among 1539 web pages, 1059 are labeled with one genre, 438 with two genres, 39 with three genres and 3 with four genres [28]; see Table II.

Table II
Composition of 20-Genre dataset

Genre	# web pages	Genre	# web pages
Blog	77	Gateway	77
Children	105	Index	227
Commercial	121	Informative	225
Community	82	Journalistic	186
Content delivery	138	Official	55
Entertainment	76	Personal	113
Error message	79	Poetry	72
FAQ	70	Adult	68
Shopping	66	Prose fiction	67
User input	84	Scientific	76

3) KRYS-I

This dataset was built between 2005 and 2008 by Kim and Ross [15]. It consists of 6494 PDF documents labeled independently by two kinds of people (students and secretaries). Each document was assigned to 1, 2 or 3 genres. A set of 70 genres has been defined, which can be classified into 10 groups (Book, Article, Short Composition, Serial, Correspondence, Treatise, Information Structure, Evidential Document, Visually Dominant Document and Other Functional Document). After removing inaccessible documents, we obtain 5339 documents (See Table III).

Table III
Composition of KRYS-I dataset

Genre	# docs	Genre	# docs
Resume/CV	124	Menu	102
Speech transcript	119	Comics	110
Poems	64	Essay	305
Chart	154	Catalog	143
Technical manual	168	Artwork	65
Memo	146	Abstract	155
Dramatic script	57	Financial record	130
Sheet music	39	Miscellaneous report	307
Research article	323	Fact sheet	228
Advertisement	103	Slides	132
Periodicals	128	Interview	107
Project description	130	Manual	237
Magazine article	254	Product description	206
Thesis	200	Forum discussion	124
Bibliographical sketch	113	Receipt	101
Letter	177	Technical report	256
Poetry book	121	Journals	139
Table calendar	108	Handbook	365
Diagram	149	Project proposal	129
Raw data	179	Minutes	116
Regulations	156	Form	271
Telegram	13	Book of fiction	24
Operational report	218	List	100
Legal proceedings	129	Contract	58
Questionnaire	135	Guideline	223
Other research article	475	Other book	62
Email	120	Slips	20
Review	198	Announcement	79
Conference proceedings	148	Appeal propaganda	40
Graph	114	Fictional piece	48
Poster	160	Order	42

B. Experimental setup

We only consider 2-grams, 3-grams and 4-grams as candidate *n*-grams since they can capture both sub-word and inter-word information. Moreover, to keep the dimensionality of the problem in a reasonable level, we removed character n-grams that appear in less than 5 web page URLs. Table IV shows the number of words and character n-grams extracted from the datasets KI-04, 20-Genre and KRYS-I.

Table IV
Number of words and 2-4 grams extracted from KI-04, 20-Genre and KRYS-I datasets

	# words	# 2grams	# 3grams	# 4grams
KI-04	120	120	117	229
20-Genre	139	185	210	317
KRYS-I	230	203	236	345

ISSN 2395-8618

The experimentation of our approach is conducted using Naïve Bayes, IBk, J48, and SMO classifiers implemented in the Weka toolkit. Due to the small number of web pages in each genre, we followed the 3-cross-validation procedure, which consists of randomly splitting each dataset into three equal parts. Then we used two parts for training and the remaining one part for testing. This process is performed three times and the final performance in terms of micro-averaged accuracy is the average of the three individual performance figures.

It is worth noting that in this study we evaluated our method using English web pages. Moreover, since character n-grams are language independent, our method can be used to classify non-english web pages.

C. Results and discussion

To evaluate our approach, we conducted two experiments. The aim of the first experiment is compare the classification accuracy obtained using words and character n-grams. While, the second experiments aims to identify the more suitable values of weighting parameters used to achieve the best accuracy.

1) Experiment 1

In this experiment we evaluated our approach using two kinds of features: words and 2-4 grams. Note that in this experiment the weighting parameters are equal to 1. Table V illustrated the achieved results for different datasets and machine learning techniques.

TABLE V
CLASSIFICATION ACCURACY USING DIFFERENT MACHINE LEARNING TECHNIQUES AND EXPLOITING WORDS AND CHARACTER *N*-GRAMS

	Words			2-4 grams		
	KI-04	20-Genre	KRYS-I	KI-04	20-Genre	KRYS-I
NB	0.71	0.64	0.63	0.72	0.73	0.68
KNN	0.75	0.55	0.62	**0.87**	**0.88**	0.82
SVM	**0.77**	**0.81**	**0.73**	**0.87**	**0.88**	**0.84**
DT	0.64	0.62	0.59	0.66	0.70	0.57

It is clear from Table V that using 2-4 grams we can achieve better results than using words. However, the best result is reported by the SVM classifier followed by KNN, Naïve bayes and then decision trees (DT). Using character 2-4 grams, the SVM achieves the highest accuracy values of 0.87, 0.88 and 0.84 for KI-04, 20-Genre and KRYS-I datasets respectively.

As shown in Table V, the overall accuracy differs from dataset to another. Using the KRYS-I dataset we obtained the lowest accuracy, while the 20-Genre dataset achieves the highest accuracy. This performance variation is due to the huge number of overlapping genres and generic web pages.

As noted by Meyer and Stein [19], multi-genre web pages are web pages where two or more genres overlap without creating a specific and more standardized genre. For example in KI-04 dataset, the genres shop and portrayal overlap. In KRYS-I dataset, the genres manual, technical manual and guideline overlap. The 20-genre dataset reports the highest accuracy because it contains few number of overlapping genres and generic web pages. Therefore, it is clear that using datasets with a big number of generic or noise web pages and overlapping genres reduces the classification performance.

As illustrated in the Table V, the SVM reported the highest accuracy due to its less over-fitting and its robustness to noise web pages. However, SVM method is more complicated and runs slowly [10, 26].

2) Experiment 2

This experiment aims to identify the appropriate values of the weighting parameters in order to achieve the best performance. In this experiment we used SVM as a machine learning technique and 2–4 grams as a classification features. The reported results are shown in Table VI.

TABLE VI
PERFORMANCE WITH DIFFERENT CONFIGURATIONS

α	β	λ	KI-04	20-Genre	KRYS-I
0	0	1	0.77	0.56	0.66
0	1	0	0.85	0.89	0.82
1	0	0	0.79	0.85	0.80
1	1	1	0.87	0.88	0.84
1	2	3	0.62	0.68	0.84
1	3	2	0.65	0.72	0.86
2	1	3	0.66	0.73	0.70
2	**3**	**1**	**0.88**	**0.90**	**0.86**
3	1	2	0.73	0.88	0.84
3	2	1	0.71	0.72	0.78

As shown in Table VI, it is clear that the segment DOCP captures more information about the genre of the web page than the segments DOCN and DOMN. Therefore, assigning the highest weight to the DOCP segment, followed by DOCN segment, then DOMN segment achieves the best result. From our experiment, the best result is reported using the values of 2, 3 and 1 for the weighting parameters α, β and λ, respectively.

V. CONCLUSION AND FUTURE WORK

In this paper, we suggested a method for genre classification of web pages that uses character n-grams extracted only from the URL of the web page. Hence, our method eliminates the necessity of downloading the content of the web page because. Moreover, our method uses a new weighting technique based on URL segmentation. Conducted experiments using three known datasets show that our method provides encouraging results. As future work, we plan to deal with generic and overlapping genres by proposing a multi-label classification where a web page can be assigned to more

ISSN 2395-8618

than one genre. Moreover, we plan to evaluate our approach using multi-lingual datasets with large number of examples.

ACKNOWLEDGMENT

Author would like to thank the anonymous reviewers for their suggested comments to improve the quality of this manuscript.

REFERENCES

[1] E. Baykan, M. Henzinger, L. Marian, I. Weber, "Purely URL based topic classification," in *Proceedings of the 18th International Conference on World Wide Web*. Madrid, Spain, 2009.

[2] T. Berners-Lee, R. T. Fielding, L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax," *Internet Society*. RFC 3986, STD 66.

[3] E. S. Boese, (2005). *Stereotyping the web: Genre Classification of Web Documents*. M.Sc. Dissertation. Colorado State University, USA, 1998.

[4] E. S. Boese, A. E. Howe, "Effects of web document evolution on genre classification," in *Proceedings of the CIKM'05*, 2005.

[5] M. Craven, D. DiPasque, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery, "Learning to extract symbolic knowledge from the World Wide Web," in *Proceeding of the 15th national / 10th conference on artificial intelligence / innovative applications of artificial intelligence*, Madison, 1998.

[6] N. Dewdney, C. Vaness-Dikema, and R. Macmillan, "The form is the Substance: Classification of Genres in Text," in *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics and 10th Conference of the European Chapter of the Association for Computational Linguistics*, Toulouse, France, 2001.

[7] A. Finn and N. Kushmerick, "Learning to classify documents according to genre," in *Proceedings of the Workshop Doing it with Style: Computational Approaches to Style Analysis and Synthesis*, held in conjunction with IJCAI 2003, Acapulco, Mexico, 2003.

[8] C. Jebari, *Une nouvelle approche de catégorisation flexible et incrémentale de pages web par genres*. Ph.D. Dissertation, Tunis El Manar University, Tunisia, 2008.

[9] C. Jebari, W. Arif, "A Multi-label and Adapative Genre Classification of Web Pages," in *Proceedings of ICMLA*, 2012.

[10] T. Joachims, "Text categorization with support vector machines: learning with many relevant features," in *Proceedings of 10th European Conference on Machine Learning*, 1998.

[11] I. Kanaris and E. Stamatatos, "Learning to recognize webpage genres," *Information Processing and Management Journal*, 45 (5) 499–512.

[12] J. Karlgren, "Stylistic experiments in information retrieval," in *Natural Language Information Retrieval* (ed. T. Strzalkowski), pp. 147–166, 1999.

[13] M. Kennedy, Shepherd, "Automatic Identification of Home Pages on the Web," in *Proc. of the 38th Hawaii International Conference on System Sciences*, 2005.

[14] B. Kessler, G. Nunberg, and H. Schütze, "Automatic detection of text genre," in *Proceedings of the 35th ACL / 8th EACL*, 32–38, 1997.

[15] Y. Kim and S. Ross, "Examining Variations of Prominent Features in Genre Classification," in *Proceedings of HICSS Conference*, 2008.

[16] C. S. Lim, K. J Lee, G. C. Kim, "Multiple Sets of Features for Automatic Genre Classification of Web Documents," *Information Processing and Management.* 41 (5) 1263–1276, 2005.

[17] J. Mason, *An n-gram-based Approach to the Automatic Classification of Web Pages by Genre*. Ph.D. Dissertation, Dalhousie University, Canada, 2009.

[18] Z. E. Meyer, *On Information Need and Categorizing Search.* Ph.D. Dissertation. Paderborn University, Germany, 2007.

[19] Z. E. Meyer, B. Stein, "Genre classification of web pages: User study and feasibility analysis," in *Proceedings KI 2004: Advances in Artificial Intelligence*. pp. 256–269, 2004.

[20] T. Mitchell, *Machine learning*. McGraw-Hill, 1997.

[21] M. Abramson, D. W. Aha, "What's in a URL? Genre Classification from URLs. Intelligent Techniques for Web Personalization and Recommender Systems," *AAAI Technical Report*. WS-12-09, 2012.

[22] P. N. Priyatam, S. Iyengar, K. Perumal, and V. Varma, "Don't Use a Lot When Little Will Do: Genre Identification Using URLs," in *14th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2013),* University of the Aegean, Samos, Greece; *Research in Computing Science* 70, pp. 233–243, 2013.

[23] M. Santini, *Automatic identification of genre in web pages.* Ph.D. Dissertation. Brighton University, UK, 2007.

[24] E. Stamatatos, N. Fakotakis, G. Kokkinakis, "Text Genre Detection Using Common Word Frequencies," in *Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000)*, Saarbrücken, Germany, 2000.

[25] M. A. Shepherd, C. Watters, "Evolution of cybergenre," in *Proceedings of the 31nd Hawaiian International Conference on System Sciences*, 1998.

[26] V. Vapnik, *The Nature of Statistical Machine Learning Theory*. Springer, 1995.

[27] V. Vidulin, M. Luštrek, M. Gams, "Using Genres to Improve Search Engines," in *1st International Workshop: Towards Genre-Enabled Search Engines: The Impact of Natural Language Processing*, Borovets, Bulgaria, pp. 45–51. 2007.

[28] V. Vidulin, M. Luštrek, M. Gams, "Multi-Label Approaches to Web Genre Identification," *Journal of Language and Computational Linguistics*, 24 (1) 97–114, 2009.

[29] I. Witten, E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, San Francisco, 2005.

ISSN 2395-8618

Improving Corpus Annotation Quality Using Word Embedding Models

Attila Novák

***Abstract*—Web-crawled corpora contain a significant amount of noise. Automatic corpus annotation tools introduce even more noise performing erroneous language identification or encoding detection, introducing tokenization and lemmatization errors and adding erroneous tags or analyses to the original words. Our goal with the methods presented in this article was to use word embedding models to reveal such errors and to provide correction procedures. The evaluation focuses on analyzing and validating noun compounds identifying bogus compound analyses, recognizing and concatenating fragmented words, detecting erroneously encoded text, restoring accents and handling the combination of these errors in a Hungarian web-crawled corpus.**

***Index Terms*—Corpus linguistics, lexical resources, corpus annotation, word embeddings.**

I. Introduction

STATISTICAL methods of natural language processing rely on large written text corpora. The main source of such texts is the web, where the amount of user-generated and social media contents are increasing rapidly. This phenomena and the structure of web contents and their production strategies result in large, but often very noisy text collections. These corpora are not only full of non-standard word forms, but also HTML entities, encoding errors, and deficient written language use (such as the complete lack of accents in texts written in a language with an accented writing system). Thus, even simple preprocessing tools, for example a custom tokenizer, might fail to process these texts correctly. One solution to this problem would be the adaptation of these tools to these specific cases and their careful application to such noisy texts. However, it is more often the case that web-crawled texts are collected in large quantities, sometimes for several different languages in parallel, and there is no time or satisfactory language competence to tune these general preprocessing tools. Moreover, if these texts are to be analyzed more deeply, the errors propagate through the whole processing chain, and become uncontrollable. The other solution to this problem is to apply a postcorrection method that is able to detect and if possible, correct these errors due to the nature of the source of the texts.

Manuscript received on March 18, 2016, accepted for publication on June 17, 2016, published on June 25, 2016.

Attila Novák is with MTA-PPKE Hungarian Language Technology Research Group, Pázmány Péter Catholic University, Faculty of Information Technology and Bionics, 50/a Práter street, 1083 Budapest, Hungary (e-mail: novak.attila@itk.ppke.hu).

In this paper, we propose a method for discovering and categorizing corpus annotation errors. We were provided with a large web-crawled corpus in Hungarian. However, its quality fell short of expectations. Since the size of the corpus was about 3 billion tokens, the errors became apparent only when it was used for a certain task, i.e. building word embedding models. This lead to the idea to use these models to detect and correct erroneous parts of the corpus. We have finally built a corpus cleaning chain handling different deficiencies of the corpus.

The paper is structured as follows. First, a brief introduction to related problems and to word embedding models is presented, which were used in several further processing steps. Then, in Section IV, our method for detecting and correcting tokenization problems is described. This step is followed by a method for detecting and restoring accents in unaccented portions of the corpus. In Section VI, we propose a method for identifying bogus noun compound analyses to eliminate annotation errors introduced by morphological analysis.

II. Related Work

The main source of noise in a web-crawled corpus is the collection of texts from social media sites. These types of texts form an independent area of research. Regarding normalization, a couple of studies aim at normalizing non-standard word usage in these texts trying to compare and convert them to wordforms corresponding to orthographic standards [1], [2] . Most recent methods published for such tasks use word embedding space transformations to estimate the normalized form of a non-standard word by finding the nearest wordform in the space of standard words to the transformed vector from the space of non-standard words [3].

Another type of problem we try to handle is the correct analysis of noun compounds. This problem is independent from the quality of the original corpus, it is present at the level of morphological analysis. In Hungarian, similarly to German, noun compounds can be built by combining nearly any nouns. However, some wordforms can be misleading, because inflected forms or even lemmas may have a grammatically possible but nonsensical compound analysis. The morphological analyzer is not able to decide in such cases what the correct decomposition of such forms are, except when the compound is explicitly included in the lexicon of the analyzer. The corpus contains a great amount of composite word forms to which the analyzer applies

ISSN 2395-8618

productive compound analysis. Dima et al. [4] have proposed a method for interpreting noun compounds for English using word embeddings. However, neither the inflectional ambiguity nor the segmentation problem are present in their case, only the task of interpretation is addressed. Checking the validity of single word compounds is similar to that of detecting multiword expressions and exploring their compositionality [5]. These studies, however, aim at determining the level of compositionality in the already identified multiword expressions. These approaches do not deal with ambiguity and the detection of real and unreal compounds. Nevertheless, we also relied on compositionality measures to evaluate possible compound analyses in the algorithm presented in this paper.

III. Word embedding models

We built two types of models using the `word2vec`[1] tool, a widely-used framework for creating word embedding representations [6], [7]. This tool implements both skip-gram and CBOW (continuous bag of word) models, generally used for building the embedding vectors. As the CBOW model has proved to be more efficient for large training corpora, we used this model. In our models, the radius of the context window used at the training phase of the model was set to 5 and the number of dimensions to 300.

Then we applied different types of preprocessing to the corpus in order to adapt the method to the agglutinating behavior of Hungarian (or to any other morphologically rich language having a morphological analyzer/tagger at hand). First, we built a model from the tokenized but otherwise raw corpus. This model assigns different vectors to different suffixed forms of the same lemma, while homonymous word forms share a single vector representation. In the other model we used a part-of-speech tagged and disambiguated version of the corpus. This was done using the PurePos part-of-speech tagger [8], which utilizes the Humor morphological analyzer [9], [10], [11] for morphological analysis and also performs lemmatization. We built a model in which each word in the corpus is represented by the sequence of two tokens: one consisting of the lemma and the other of the morphological tags generated by the tagger/morphological analyzer like in [12].

When using the models built from the raw and the annotated corpus for other tasks, different types of errors were revealed when investigating lists of similar words for certain seed terms. These were the following: (1) simple misspellings, (2) unaccented forms, (3) forms with character encoding errors, (3) word fragments, (4) annotation errors. Even though these erroneous forms were to some extent also semantically related, the errors often overshadowed semantic similarity and words with the same error type were clustered by the models.

Performing deeper analysis regarding the source of these errors lead us to the inadequate quality of the original corpus. However, we could rely on the embedding models created from the erroneous corpus to implement methods aiming at improving the quality of the corpus and its annotation.

[1] https://code.google.com/p/word2vec/

IV. Detecting word fragments and tokenization errors

One of the problems the models revealed was the presence of a great number of word fragments. These were for the most part introduced by the custom tokenizer applied to the texts. Fortunately, the tokenizer inserted a glue tag `<g/>` at places where it broke single words. The glue tag indicates that there was no whitespace at the position of the tag in the original text. Examples for such situations are hyphens or other punctuation marks, numbers within words, or HTML entities within words. However, some of these splits were nonsense, for example if an HTML entity within a word indicated possible hyphenation boundaries, but not real segmentation, then these words were split at all such boundaries. The tokenizer also segmented words at HTML entities encoding simple letter characters. Fortunately, these erroneous splits could be undone by finding glue tags in contexts where they should not occur.

However, not all word splits were explicitly marked with these tags. If an HTML tag was inserted into a word, then the word was simply split at these points, leaving no track of its original form. These could only be tracked by finding the original HTML source of the texts.

Another source of seemingly fragmented words was due to incorrect lemmatization. These forms appeared only in the model built from the analyzed corpus and could be identified by looking them up in the embedding model of surface forms. If a fragment appeared only in the analyzed model, then it was a lemmatization error.

In order to measure the relevance of this error in the corpus, and the proportion of the various causes, we collected a set of word fragments from the corpus. This could be done easily by querying the embedding models for the nearest neighbors of some fragments. Such queries resulted in lists of fragments hardly containing any real words. These real words could then be easily filtered out by clustering the resulting set. The hierarchical clustering algorithm we applied, grouped real words into a few distinct clusters. Projecting this initial set of fragments to the whole vocabulary revealed that 3.7% (!) of the most frequent 100,000 noun, adjective and verb “lemmas” identified by the annotation system was due to the presence of such fragments in the corpus.

Revisiting the glue tags introduced by the tokenizer and unifying those words that should not have been split at this stage corrected 49.36% of these errors. Then, since these fragments were collected from the analyzed model (due to its more robust and coherent representation of words), fragments in the remaining list were checked in the embedding model of surface forms in order to eliminate the errors introduced by the lemmatizer. This method revealed that 17.08% of the original list originated here. This result can be a good starting point for improving the accuracy of the lemmatizer in PurePos

ISSN 2395-8618

TABLE I
SUMMARY OF RESULTS REGARDING WORD FRAGMENTS IN THE CORPUS

type	**percentage of tokens**
original state	3.70%
corrected state	1.04%
type	**percentage of word fragments**
incorrect splits indicated by glue tags	49.36%
lemmatization errors	17.08%
validated by a spell checker	5.45%

used for words not analyzed by the morphological analyzer. However, handling that problem is out of the scope of this paper. Since a major part of these lemmatization errors is due to spelling or capitalization errors in the original corpus, which resulted in the failure of morphological analysis and the lemmatizer guesser being applied, most of these errors should be handled by identifying and correcting the errors in the corpus. A further 5.45% of the list was validated by a spellchecker as correct word form. However, this did not mean that these strings could not be fragments of longer words at the same time. Thus, 71.89% of these fragmentation errors could be eliminated, reducing the original percentage of 3.7% to 1.04%. These remaining errors are mostly due to the incorrect parsing of the HTML source, splitting words in case of HTML tags within words, without leaving any trace of doing this. Since we had no access to the original HTML source of the corpus, we could not correct these errors. Table I summarizes the results.

V. RESTORING ACCENTS

In Hungarian, umlauts and acute accents are used as diacritics for vowels. Acute accents mark long vowels, while umlauts are used to indicate the frontness of rounded vowels *o*→*ö* [o→ø] and *u*→*ü* [u→y], like in German. A combination of acutes and umlauts is the double acute diacritic to mark long front rounded vowels *ő* [øː] and *ű* [yː]. Long vowels generally have essentially the same quality as their short counterpart (*i-í, ü-ű, u-ú, ö-ő, o-ó*). The long pairs of the low vowels *a* [ɔ] and *e* [ɛ], on the other hand, also differ in quality: *á* [aː] and *é* [eː]. There are a few lexicalized cases where there is a free variation of vowel length without distinguishing meaning, e.g. *hova*∼*hová* 'where to'. In most cases, however, the meaning of differently accented variants of a word is quite different. Table II shows all the possible unaccented-accented pairs of vowels in Hungarian together with their distribution in a corpus of 1 804 252 tokens.

TABLE II
POSSIBLE ACCENT VARIATIONS IN HUNGARIAN

a	a: 70.33%; á: 29.66%
e	e: 73.40% é: 26.59%
i	i: 86.04% í: 13.95%
o	o: 55.41% ó: 14.65% ö: 15.82% ő: 14.10%
u	u: 46.96% ú: 12.72% ü: 29.98% ű: 10.32%

Due to their meaning distinguishing function, it is crucial for any further processing steps to have the accents in the texts.

TABLE III
RATIO OF EACH LANGUAGE/TEXT TYPE IN THE CORPUS

Language/type	**Number of words**	**Percentage**
All	2684584137	100.00%
Hungarian	2560265742	95.37%
Encoding error	88668867	3.30%
Unaccented Hungarian	9177770	**0.34%**
English	7535446	0.28%
German	4202044	0.16%
French	767515	0.03%
Latin etc.	1311286	0.05%
Short rest	12655467	0.47%

However, due to the widespread use of smart mobile devices, more and more texts on the web are created without accents, because these devices do not really provide a comfortable and fast possibility to type accented characters. The embedding models used in our experiments also justified this assumption, generating unaccented forms as nearest neighbors for some seed words. In order to detect such portions of the corpus, we trained the TextCat language guesser [13] on standard and unaccented Hungarian. We also used language models for other languages we identified as being present in the corpus with this tool to categorize each paragraph of the original corpus. Furthermore, two more categories were also considered, namely encoding errors and short paragraphs (the language of which cannot be reliably identified by TextCat). Erroneous identification of the source code page of HTML pages resulted in encoding errors, which often also resulted in fragmentation of words by the tokenizer. Even though compared to the size of the whole corpus, the amount of text written in other languages, missing accents or being erroneously decoded does not seem to be too much, errors of this type affect the vocabulary present in the corpus significantly because even an erroneous subcorpus of a size of a couple of 10 million tokens results in a million of erroneous word types injected into the vocabulary.

The results are shown in Table III.

As it can be seen from the percentages, the ratio of texts containing encoding errors and unaccented texts is quite high.

Once recognized, unaccented paragraphs can be corrected by applying an accent restoration system. We used the one described in [14], a system based on an SMT decoder augmented with a Hungarian morphological analyzer. Since we had access to that system and to the model built for the experiments described in the paper, we did not have to train, but could just use the system as it was. This tool could restore accented words with an accuracy above 98%.

VI. CORRECTING BOGUS NOUN COMPOUND ANALYSES

In Hungarian, noun compounds are very frequent. The most productive compounding pattern is concatenating nouns. In many cases certain inflected forms can also be analyzed as a compound. In such cases the morphological analyzer is not able to choose the correct segmentation unless the compound

ISSN 2395-8618

TABLE IV
EXAMPLES FOR AMBIGUOUS SEGMENTATIONS OF WORDS. THE CORRECT (OR MORE PROBABLE) ONES ARE TYPESET IN BOLDFACE.

original form	possible segmentation	meaning in English
gázló	gáz+ló N	'gas+horse'
	gázló N	'ford'
tűnő	tű+nő N	'needle+woman'
	tűnő V.PrtPres	'looking like sg.'

is explicitly included in the lexicon of the analyzer. Some examples for ambiguous segmentation are shown in Table IV

Although the lexicon of the morphological analyzer contains many compound stems, nevertheless in a big corpus there will always be words where productive compounding is needed to yield a valid analysis. Moreover, although many bogus compound analyses are prevented in the analyzer by excluding certain nouns from compounding, productive compounding may still result in bogus compound analyses. Thus, handling this very elemental problem can also be considered a corpus quality issue, because morphological analysis is the basis of many other NLP tasks. And again, we used word embedding models to create a method for identifying erroneous compound segmentation. The morphological analyzer used in our experiments [9], [10] is able to suggest the various possible segmentations, but is not able to choose the correct one. The problem to be solved can be considered a binary classification problem, in which the algorithm has to decide whether a segmentation candidate is correct or not.

First, all words from the corpus were selected for which the morphological analyzer suggested at least one productively generated compound segmentation (either correct or incorrect). From this list of 6,122,267 words, a random selection of 1300 words were taken apart for development and testing purposes. This set was manually checked and the correct segmentations were chosen.

We created one baseline system that queried all possible compound members for all analyses returned by the morphological analyzer, and sorted them by their similarity to the original word form in the vector space model generated from the raw corpus. For compounds consisting of more than two elements, all compound member sequences that did not match the whole stem were also included in the list. We then selected the top-ranked item in this list (the one closest in the vector space to the original word form), and excluded all analyses which were not compatible with this item. If the top-ranked item matched a lexically given segmentation in the lexicon of the morphological analyzer, we accepted that segmentation. All analyses not excluded by the top-ranked item were kept as possible ones.

In the other system, several features were determined for each word for each segmentation suggested by the analyzer. First, the constructing elements of the actual segmentation were ranked according to their similarity to the original form, for which the similarity values were extracted from the embedding model (this step corresponds to the first baseline system). In addition, assuming that the meaning of compounds should be compositional, the 10 nearest neighbors for each element were also retrieved from the embedding model, and all of these were combined using the segmentaton of the original word as a model, producing analogous variants for the original word where compound members are substituted with synonymous words. This list of analogous words was then also ordered by each item's similarity to the original word. Having these ordered lists, the following numerical features were derived:

- A: The similarity of the first-ranked element of the original segmentation
- B: The average of the similarities of all elements of the original segmentation
- C: The similarity of the first-ranked analogous variant (or zero, of no analogous variant was found)
- D0: The length of the list of analogous variants with similarity greater than zero
- D1: The average of the similarities of analogous variants with similarity greater than zero
- D2: D0*D1
- D3: The average of D0, D1 and C

Once these features were extracted, a simple binary decision tree was trained for each of these features individually and for the combination of all of these features. For training and testing, we applied a 10-fold crossvalidation using the previously separated and manually labelled list of 1200 words with a different 9:1 split in each run. The results are shown in Table V. The table contains the accuracy of each system, i.e. the ratio of correctly predicting the correctness of a given segmentation for a certain word. As it can be seen from the table, the most significant feature turned out to be the length of the list of analogous variants. This suggests, that if there is a large enough number of words created by substituting each element of a proposed segmentation with words of similar meaning and the resulting compositions are existing words, then the segmentation can be considered as a valid compound with almost 90% certainty. While the first baseline system relied on lexical knowledge embodied in the compound analyses listed in the lexicon of the morphological analyzer, the decision-tree-based systems did not use that information. The success of the D0 system seems to indicate that compositionality and variability is an important property of productive compounding.

Thus, integrating this feature into the compounding model implemented in the morphological analyzer can also have a beneficial effect on the quality of the annotation.

VII. CONCLUSION

In this paper we explored methods based on continuous vector space models that can be used to identify and correct errors in corpus annotation ranging from errors resulting from erroneous language identification or encoding detection through tokenization and lemmatization errors to erroneous

ISSN 2395-8618

TABLE V
THE PRECISION OF EACH SYSTEM CREATED FOR VALIDATING CORRECT SEGMENTATIONS OF POSSIBLE COMPOUNDS

System	Precision
first baseline	86.45%
decision tree for feature A	82.32%
decision tree for feature B	82.41%
decision tree for feature C	85.17%
decision tree for feature D0	**90.34%**
decision tree for feature D1	85.43%
decision tree for feature D2	84.22%
decision tree for feature D3	85.43%
decision tree for all features	85.34%

compound analyses. As these models effectively map tokens having a similar distribution to similar locations in vector space, they can be used to retrieve and cluster tokens in the corpus that are there due to the same types of errors in the annotation tool chain revealing the nature and the possible source of these error. Moreover, the distributional models can also be used to identify possible errors in the annotation such as bogus compound analyses exploiting the fact that productive compounding is in general a compositional operation. Here we did not explore the possibility of taking advantage of these models for the identification and correction of errors inherently present in the corpus, such as spelling errors. Nevertheless, that seems to be another promising application area.

REFERENCES

[1] V. K. Rangarajan Sridhar, "Unsupervised text normalization using distributed representations of words and phrases," in *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*. Denver, Colorado: Association for Computational Linguistics, June 2015, pp. 8–16.

[2] C. Li and Y. Liu, "Improving text normalization via unsupervised model and discriminative reranking," in *Proceedings of the ACL 2014 Student Research Workshop*. Baltimore, Maryland, USA: Association for Computational Linguistics, June 2014, pp. 86–93.

[3] L. Tan, H. Zhang, C. Clarke, and M. Smucker, "Lexical comparison between Wikipedia and Twitter corpora by using word embeddings," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. Beijing, China: Association for Computational Linguistics, July 2015, pp. 657–661. [Online]. Available: http://www.aclweb.org/anthology/P15-2108

[4] C. Dima and E. Hinrichs, "Automatic noun compound interpretation using deep neural networks and word embeddings," in *Proceedings of the 11th International Conference on Computational Semantics*. London, UK: Association for Computational Linguistics, April 2015, pp. 173–183.

[5] B. Salehi, P. Cook, and T. Baldwin, "A word embedding approach to predicting the compositionality of multiword expressions," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Denver, Colorado: Association for Computational Linguistics, May–June 2015, pp. 977–983. [Online]. Available: http://www.aclweb.org/anthology/N15-1099

[6] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 3111–3119.

[7] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *CoRR*, vol. abs/1301.3781, 2013.

[8] G. Orosz and A. Novák, "PurePos 2.0: A hybrid tool for morphological disambiguation," in *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2013)*. Hissar, Bulgaria: INCOMA Ltd. Shoumen, BULGARIA, 2013, pp. 539–545.

[9] A. Novák, "Milyen a jó humor? [What is good humor like?]," in *I. Magyar Számítógépes Nyelvészeti Konferencia [First Hungarian conference on computational linguistics]*. Szeged: SZTE, 2003, pp. 138–144.

[10] G. Prószéky and B. Kis, "A unification-based approach to morpho-syntactic parsing of agglutinative and other (highly) inflectional languages," in *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, ser. ACL'99. Stroudsburg, PA, USA: Association for Computational Linguistics, 1999, pp. 261–268.

[11] A. Novák, "A new form of humor – mapping constraint-based computational morphologies to a finite-state representation," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, N. Calzolari, K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, and S. Piperidis, Eds. Reykjavik, Iceland: European Language Resources Association (ELRA), May 2014, pp. 1068–1073.

[12] B. Siklósi, "Using embedding models for lexical categorization in morphologically rich languages," in *Computational Linguistics and Intelligent Text Processing: 17th International Conference, CICLing 2016, Konya, Turkey, April 3-9, 2016*, A. Gelbukh, Ed. Cham: Springer International Publishing, 2016.

[13] K. Hornik, P. Mair, J. Rauch, W. Geiger, C. Buchta, and I. Feinerer, "The textcat package for n-gram based text categorization in R," *Journal of Statistical Software*, vol. 52, no. 6, pp. 1–17, 2013.

[14] A. Novák and B. Siklósi, "Automatic diacritics restoration for hungarian," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, September 2015, pp. 2286–2291. [Online]. Available: http://aclweb.org/anthology/D15-1275

ISSN 2395-8618

A Method Based on Patterns for Deriving Key Performance Indicators from Organizational Objectives

Carlos Mario Zapata Jaramillo and Luis Fernando Castro Rojas

***Abstract*—Organizational strategic alignment implies consistency among the organizational elements. Organizational objectives act as essential elements for leading such alignment. In addition, key performance indicators (KPIs) have demonstrated usefulness for assisting the strategic alignment allowing for a holistic control of the organization. Some approaches emphasizing objective-KPI relationships have been proposed; however, they lack of a fully appropriate method for treating organizational objectives, KPIs, and objective-KPI relationships. They exhibit some drawbacks in terms of ambiguity, stakeholder understandability, and subjectivity. In this paper, we propose a method for overcoming such drawbacks, by using pre-conceptual-schema-based organizational patterns as a way to operationalize organizational objectives in terms of the KPIs. So, a systematic method for deriving a set of candidate KPIs from a specific organizational objective is provided. In addition, we present a lab study in order to illustrate the main aspects of this proposal.**

***Index Terms*—Strategic alignment, organizational objective, key performance indicator, pattern.**

I. Introduction

STRATEGIC alignment for improving the performance of an organization has received a renewed attention among academics and practitioners since the 1990s [1]. Some authors conclude that means for establishing and assessing such alignment are underdeveloped [2]. Consequently, achieving this alignment will be one of the most important management priorities in the upcoming years [3].

Several studies agree that organizational objectives are essential for leading the strategic alignment [2–5]. According to Basili *et al*. [4], such objectives are not always explicitly nor clearly stated in the organizations. Thus, the verification of the achievement of those objectives is a difficult task.

Some other studies propose key performance indicators as vital factors for guaranteeing strategic alignment [2], [4], [5]. According to Cosenz [6], "what cannot be measured cannot be controlled." Consequently, strategies and assessment practices, such as the adoption of proper performance indicators, need to be implemented. In addition, according to Kronz [7], the holistic management can be achieved by collecting and analyzing performance in terms of key performance indicators (KPIs).

As a result, relationships among organizational objectives and key performance indicators are becoming more important. According to Basili *et al*. [5], an integrated vision of the company can be obtained by aligning and communicating all the goals, strategies, and measurement opportunities. So, all of the organizational elements can be directed to the same target. In most of the companies, performance measurement indicators have been developed; however, little effort to adequately establish links between objectives and performance indicators has been devoted [8].

Some approaches for exploring organizational objectives, KPIs, and objective-KPI relationships have been proposed. Some of them have metrics and models for measuring objective achievement. For example, Shamsaei *et al*. [9] propose an approach for modeling the context and measuring compliance levels based on certain rules. Pourshahid *et al*. [10] describe a method for decision making by including objectives, decision-making devices, and KPIs. Barone *et al*. [11] and Maté *et al*. [12] outline a method for modeling processes, objectives, indicators, and situations affecting the objectives. Strecker *et al*. [13] present a conceptual modeling proposal with the aim of satisfying essential requirements in the domain of organizational performance measurement. Frank *et al*. [14] outline a method for modeling indicator systems, based on a modeling language called SCORE-ML. Some proposals are mainly theoretical, and specifications for guiding any computational tractability are ignored, e.g., BSC (Balanced Scorecard) [15] and GQM (Goal Question Metrics)+ Strategies® [4], [17]. Thevenet [18] proposes the INSTAL method, which is aimed at aligning

Manuscript received on February 27, 2016, accepted for publication on May 13, 2016, published on June 25, 2016.

Carlos Mario Zapata Jaramillo is with the Facultad de Minas of the Universidad Nacional de Colombia, Colombia (e-mail: cmzapata@unal.edu.co).

Luis Fernando Castro Rojas is with the Facultad de Ingeniería of the Universidad del Quindío and the Facultad de Minas of the Universidad Nacional de Colombia, Colombia (e-mail: lufer@uniquindio.edu.co, lufcastroro@unal.edu.co).

ISSN 2395-8618

information systems to strategic business objectives in an organization. Doumi *et al*. [19] propose the modeling of strategic alignment among objectives and information systems by using indicators. Giannoulis *et al*. [2], [20] present the SMBSC (strategic maps and balanced scorecard) metamodel, which includes the concept of measure for supporting the evaluation of an objective performance. Finally Popova *et al*. [21] present a formal framework for modeling objectives, performance indicators and their relationships.

Unfortunately, the aforementioned contributions lack a wholly appropriate method for treating organizational objectives, KPIs, and objective-KPI relationships. They exhibit some drawbacks: i) some proposals are used for expressing objectives and KPIs by using informal natural language, causing ambiguity problems; ii) other proposals are used for expressing objectives and KPIs too formally inducing stakeholder understandability problems; and iii) in some cases the objective-KPI relationships are established by using personal criteria involving subjectivity problems. In the next Section we present a comparison of such contributions in order to evidence the aforesaid drawbacks.

In this paper we propose a novel method based on patterns for overcoming the previously identified drawbacks. Such a method allows us for deriving a set of candidate KPIs from organizational objectives. The method is based on pre-conceptual schemas [22], [23] for representing organizational objectives, key performance indicators, objective-KPI relationships, and the involved domain elements. Also, it uses a set of patterns for deriving candidate KPIs. As a result, a lab study is developed in order to demonstrate how the key performance indicators can be derived from organizational objectives by identifying and by applying certain patterns.

This paper is organized as follows: in Section II we show a review of the related work, and we present additional background information for facilitating a comparison among several approaches dealing with organizational objectives, KPIs, and objective-KPI relationships. In Section III we describe the proposal based on patterns for deriving key performance indicators from organizational objectives. In Section IV we provide a lab study for illustrating the main aspects of the proposal. Finally, in Section V we present some conclusions and future work.

II. Related Work

A. Objective-Oriented Proposals

Several approaches have been proposed for studying the strategic alignment in organizations. More specifically, such approaches have been focused on organizational objectives, KPIs, and objective-KPI relationships.

A first set of approaches based on GORE (goal-oriented requirements engineering) establishes the objectives as main components for contributing to the strategic alignment. Such proposals only emphasize on representing, defining, and analyzing the organizational objectives, but they disregard the inclusion of KPIs and objective-KPI relationships. Some of such proposals are presented by Kavakli [24], Engelsman *et al*. [25], Yu *et al*. [26], de la Vara *et al*. [27], Gröner *et al*. [28], and Giannoulis *et al*. [29]. The latter formalize the strategy maps [30] by using a metamodel. The MAP methodology [31] allows for the modeling of the high-level goals and strategies of an enterprise.

B. KPI Oriented Proposals

Another set of proposals is focused on representing, defining, and analyzing the KPIs, but they disregard the inclusion of organizational objectives and objective-KPI relationships. Some of such proposals are presented by Caputo *et al*. [32] using SBVR (semantics of business vocabulary and business rules) for developing a KPI Vocabulary; Del-Río-Ortega *et al*. [33] proposing an ontology for the definition of process performance indicators; and Wetzstein *et al*. [34] using WSML (web service modeling language) for defining a KPI ontology.

C. Objective-KPI Oriented Proposals

Since the aforementioned set of proposals provides crucial elements for supporting the method we propose in this paper, we are mainly centered on those approaches aimed at including both organizational objectives and KPIs, as well as the objective-KPI relationships. Therefore, in this section, we review such proposals; besides, we analyze them with the previously established drawbacks in mind.

BSC (balanced scorecard) [14] includes causal linkages among objectives and performance measurements. In this context, objectives are measurable and the measures support the assessment for fulfilling such objectives.

GQM+ Strategies [4], [17] is an approach for linking organizational operational objectives and strategies from the top management level to the project level and back. Also, this approach is intended to align the business at all levels of the organization in a seamless way. Finally, GQM+ provides a mechanism for monitoring the success and failure of objectives and strategies by using measurement.

Giannoulis *et al*. [2] introduce the SMBSC meta-model, which allows for the integration of strategy maps and BSC by using a metamodel. Then, Giannoulis and Zdravkovic [35] present a case scenario based on i* and SMBSC.

Strecker *et al*. [13] introduce a conceptual model for representing an indicator system by rationalizing the process of creating, using, and maintaining indicators. Such a proposal includes the indicator-objective relation, and benefits from the reuse of MEMO (multi-perspective enterprise modeling) modeling concepts. Another proposal integrated with MEMO is outlined by Frank *et al*. [14]. The authors propose a modeling method for designing business indicator systems related to the organizational objectives. The modeling

ISSN 2395-8618

concepts are defined by using the performance modeling language called SCORE-ML.

Doumi *et al*. [19] propose the modeling of strategic alignment among objectives and information systems by using indicators. Such indicators are associated with tasks and they are used for helping stakeholders to implement actions for achieving objectives. In addition, the indicators facilitate the verification of the accomplishment of the organizational objectives and the reorganization of the business processes.

The INSTAL method introduced by Thevenet [18] aims at aligning information systems with organizational objectives. Such a method involves metrics applied to strategic elements and measures applied to operational elements. However, in this method, performance indicators are lightly treated, and explicit objective-KPI relationships are not considered.

Two extensions of URN (user requirements notation) have been proposed. Shamsaei *et al*. [9] propose some rules for modeling the context and measuring the compliance level of their process. In addition, a regulation model, including policies, sub-policies, rules, and KPIs is provided. Pourshahid *et al*. [10] propose a method for making decisions, including objectives, decision-making devices, and KPIs into a single conceptual framework. In both URN extensions, a graphic tool is used for representing and analyzing KPIs. Basic notation of such a tool is illustrated in Figure 1. Unfortunately, KPIs are subjectively related to the objectives, and the specification of KPIs is carried out in an uncontrolled way by using natural language.

Barone *et al*. [11] and Maté *et al*. [12] use BIM (business intelligence model) for modeling the business strategy. This framework comprises the modeling of objectives, indicators, and potential situations affecting objectives. Also, some techniques and algorithms are provided for deriving values and composing indicators. Such proposals are supported by a visual editor prototype for drawing business schemas and reasoning about them. An example of this framework is illustrated in Figure 2.

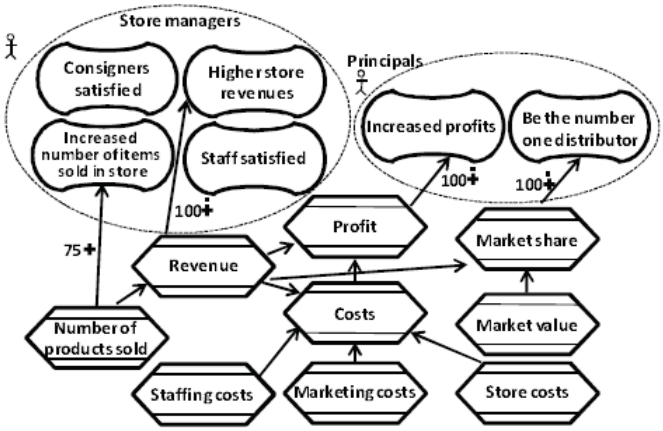


Fig. 1. Representation of KPIs and Objectives. Source: [9]

Popova *et al.* [21] present a framework for formally modeling predicate-logic-based performance indicators and their relationships by using a performance-oriented view. LEADSTO language and the LEADSTO property editor tool are used in the modeling process. LEADSTO language is a sublanguage of TTL (temporal trace language) which enables direct temporal—or causal—dependency modeling among state properties. Besides, the relationships are formally defined by using axioms expressed in TTL. The components of the framework and the considered modeling views—process, performance, organization, and agent—can be directly related to the components of the GERAM (the generalized enterprise reference architecture and methodology). This approach is illustrated in Figure 3.

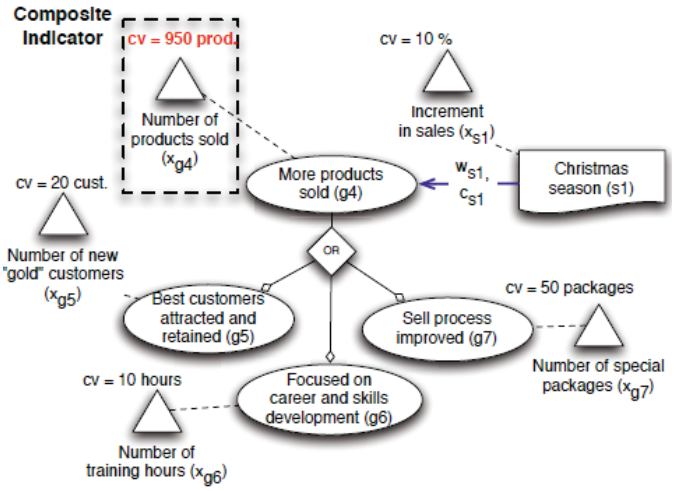


Fig. 2. Relationships among objectives and KPIs. Source: [10].

D. *Comparison of Proposals*

Table 1 shows a comparison of the aforementioned proposals. We use the following notation: "✓" means the proposal successfully addresses the issue; "p" indicates the proposal addresses it partially; and "✗" indicates the proposal does not contemplate the issue.

We see some proposals related to the objective-KPI relationships. However, none of the analyzed approaches provides an assisted method that would systematically guide an appropriate derivation of KPIs. Moreover, some of such proposals specify objectives and KPIs either informally in natural language or simply as labels within a specific diagram, leading to ambiguity problems. Other proposals express objectives and KPIs by using formal specifications that induce stakeholder understandability problems. Lastly, some proposals establish objective-KPI relationships by using only the personal criterion involving subjectivity problems. For these reasons, more research on this field is needed.

ISSN 2395-8618

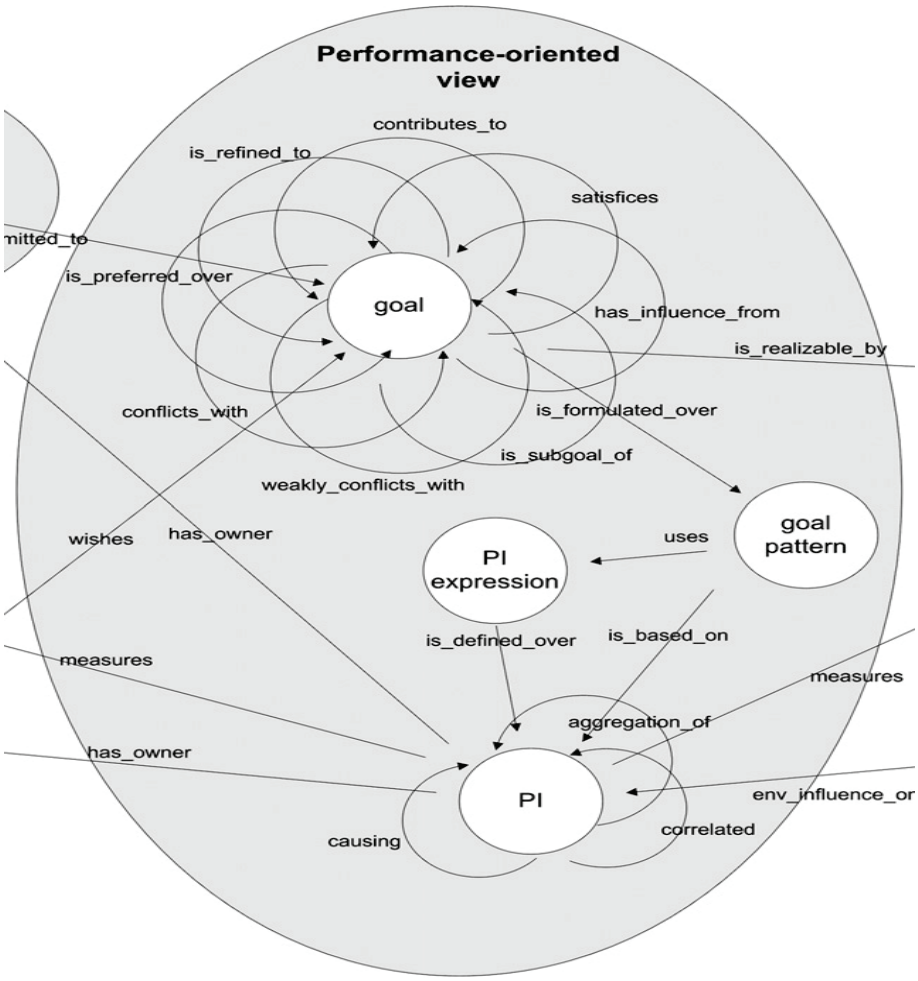


Fig. 3. Fragment of Meta-model for the performance-oriented view. Source: [21]

III. Our Proposal

The main weakness of the current research in this area is the lack of a systematic approach for deriving KPIs from the concepts involved in specific organizational objective domain. In this paper we propose a systematic method for deriving KPIs from organizational objectives. Thus, we can correctly map elements representing the organizational objective domain to elements representing the performance indicator domain. These indicators are useful for assessing the fulfillment of the organizational expectations. In addition, an appropriate traceability among the aforementioned elements can be established.

A method for systematically linking organizational objectives to KPIs should include the following features:

- The method should provide a clear understanding of the domain where the organizational objectives are defined.
- The method should support the analyst by providing techniques to perform the derivation process of KPIs in a systematic and precise way.

Accordingly, our proposal includes the following phases as a way to deal with the previously mentioned problems:

1) Defining a modeling process for describing the organizational objective and their domain.
2) Developing a methodological guideline for deriving KPIs from organizational objectives.

A. Phases

1) Modeling phase:

In this step, we represent the actual domain of the enterprise related to such organizational objective. We use the pre-conceptual schemas [22] for modeling the organizational objectives and their domain. Pre-conceptual schemas have several advantages: unambiguous syntax, integration of concepts, dynamic elements, and proximity to the stakeholder language. In addition, untrained stakeholders can understand pre-conceptual schemas [23].

ISSN 2395-8618

TABLE I
COMPARISON OF STUDIED PROPOSALS. SOURCE: THE AUTHORS

Proposal	Aspects of comparison				
	Objective modeling	KPI modeling	Objective-KPI relationship modeling	Assisted KPI derivation	Detected drawbacks
Shamsaei *et al*. [9]	✓	✓	✓	✗	Ambiguity, Subjectivity
Pourshahid *et al*. [10]	✓	✓	✓	✗	Ambiguity, Subjectivity
Barone *et al*. [11]	✓	✓	✓	✗	Ambiguity, Subjectivity
Maté *et al*. [12]	✓	✓	✓	✗	Ambiguity, Subjectivity
Doumi *et al*. [19]	✓	✓	✓	✗	Ambiguity, Subjectivity
Frank *et al*. [14]	✓	✓	✓	✗	Subjectivity
Strecker *et al*. [13]	✓	✓	✓	✗	Subjectivity
Popova *et al*. [21]	✓	✓	✓	✗	Stakeholder understandability
Thevenet [18]	✓	P	✗	✗	Ambiguity, The modeling of Objective-KPI relationships is not considered.
Del-Río *et al*. [33]	✗	✓	✗	✗	Objectives and Objective-KPI relationships are not considered in the model.
Wetzstein *et al*. [34]	✗	✓	✗	✗	Objectives and Objective-KPI relationships are not considered in the model.
Caputo *et al*. [32]	✗	✓	✗	✗	Objectives and Objective-KPI relationships are not considered in the model.

Basic Concepts and Notation

Nodes

Pre-conceptual Schemas include five kinds of nodes connected to two kinds of arcs for graphically representing a model [22, 23].

Every *concept* has an incident connection-type arc starting or ending in a relationship. The *concept* node can be of two kinds:

- *Class-Concept* is a concept that contains attributes and can be instantiable.
- *Attribute-Concept* is a leaf concept within a Pre-conceptual Schema, i.e., an attribute of a *Class-Concept*.
- A *dynamic relationship* represents actions and has exactly one incoming and one outgoing *connection* arcs. Such a relationship can be connected to concepts by using *connections arcs*. In addition, it can be connected to other dynamic relationships by using *implication* arcs.
- A *structural relationship* can be either is-a or part/whole type. It is incident to *concepts* and has exactly one incoming and one or more outgoing arcs of *connection* type.
- An *achievement relationship* can be connected to *structural* and *dynamic relationships* and *concepts*. In addition, *achievement relationships* can be connected among each other by using *implication* arcs.

Arcs

- A *connection arc* connects a *concept* to a *relationship* or vice versa.
- An *implication arc* connects a *dynamic relationship* to a *dynamic relationship*. *Implication arc* can be also used for connecting *achievement relationships* among each other.

The basic notation of the pre-conceptual schemas is illustrated in Figure 4.

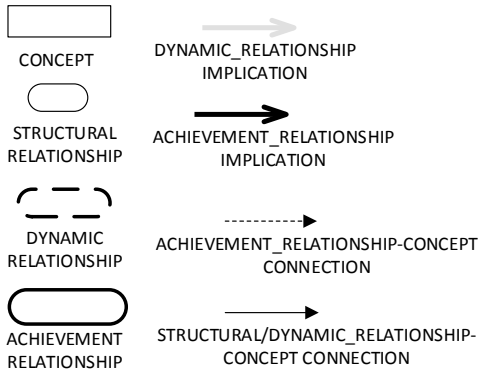


Fig. 4. Basic elements of Pre-conceptual Schemas. Source: [23]

2) Derivation Phase

In this phase, a set of candidate KPIs is derived from organizational objectives and their previously modeled domain. The derivation process comprises a set of patterns, which guides the analysis of the several possibilities for identifying key performance indicators. According to Martinez [36], a pattern reflects something to be used in a number of situations and, thus, has some generality. She establishes three features of the patterns: (1) the description of a pattern contains a context, which explains the intent of the pattern and suggests how it should be used; (2) patterns express solutions in a variety of ways, depending on the details of a circumstance; and (3) pattern descriptions can express architectural considerations, regardless of specific languages and design methodologies.

ISSN 2395-8618

Consequently, we define a set of patterns to be systematically processed and to guarantee recurrent solutions in every step of the process. Such patterns specify candidate KPIs from an organizational objective model.

B. Description of the Pattern

We use a template—based on the work developed by Martinez [36]—for describing the patterns defined in this paper. Hence, the basic elements we use are the following:

- *Name:* A sentence summarizing the pattern.
- *Context:* A situation characterizing a problem. It describes situations in which the problem occurs.
- *Problem*: The recurring problem arising in the context.
- *Structure*: A detailed specification of the structural aspects of the pattern.
- *Solution*: The strategy for solving the recurring problem.
- *Examples*: A model intended to illustrate the pattern.

C. Patterns in the organizational objective model

The patterns proposed in this paper are focused on identifying several structures belonging to an organizational objective model that can be useful for deriving some KPIs related to such organizational objective. Then, a specific pattern will be used depending on the type of structure identified.

Proposed Patterns

The KPIs derivation process is carried out in a systematic way by identifying those relevant elements in the model suggested by the stakeholder and by analyzing such elements. In order to complete such task, we provide a set of patterns based on pre-conceptual schemas for systematically guiding the derivation process. In this section we present some examples of such patterns:

The *leaf-attribute pattern*: To be applied when an *attribute concept*—which is related to an *achievement relationship*—can be used for generating a set of candidate KPIs.

The *dynamic-relationship pattern*. To be applied when a *dynamic relationship*—which is related to an *achievement relationship*—can be used for generating a set of candidate KPIs.

Applying Patterns

The derivation phase can be initiated after modeling the domain of the enterprise related to an organizational objective. Thus, the proposed patterns can be used when some part of the pre-conceptual schema matches the pattern. In order to complete this task, a specific method for applying the proposed patterns is presented. The method comprises the following steps:

Step 1. Analysis of modeled elements.

Each of the modeled elements should be analyzed in order to determine if such an element can be considered as a candidate element. A candidate element should meet the features described in any of the proposed patterns.

Step 2. Identification of the appropriate pattern.

Once a candidate element—and the portion of the pre-conceptual schema—have been identified, the suitable pattern-type should be determined.

Step 2.1. When an attribute concept is encountered and it is linked to an *achievement relationship*, the *leaf-attribute pattern* (pattern 1) can be applied.

Step 2.2. When a *dynamic relationship* is encountered and it is linked to an *achievement relationship*, the *dynamic-relationship pattern* (pattern 2) can be applied.

Step 3. Derivation of KPIs.

The procedure described in the pattern solution should be followed up in order to carry out the derivation of KPIs from the modeled organizational objective.

Catalog of Patterns

In this Section, each pattern is explained in detail by using the structure defined by Martinez [36].

1) The leaf-attribute pattern.

Context

This pattern is applied when an *attribute-concept* is candidate for deriving a set of KPIs. An example of this structure is shown in Figure 5.

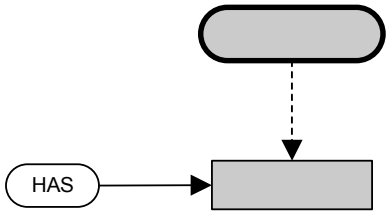


Fig. 5. Structure of leaf-attribute pattern. Source: The authors.

Problem

The problem relies on generating a set of KPIs from a candidate *attribute-concept*, which is determined by using the pattern 1. Also, the type of *achievement relationship*—improvement, maintenance, or accomplishment—linked to the concept should be identified. So, the appropriate KPI structure can be defined.

ISSN 2395-8618

Structure

The elements involved in this pattern are as follows:

Attribute-concept. The concept used for deriving the set of KPIs.

Achievement relationship. One of the relations linked to the concept. Such a relationship can be improvement-, maintenance-, or accomplishment-type.

Structural relationship. One of the relations linked to the concept. The type of this relationship is part/whole.

Solution

In this paper, the *achievement relationships* involved in the case study are treated as *improvement*-type. More detailed information about *achievement-type relationships* is provided by Lezcano [16]. The process for applying this pattern is as follows.

A KPI is obtained by writing a function name from the list: *amount of, average, maximum, minimum, number of*. Then, the label of the concept is added. Also, in case of *average* function, the KPI is formed as follows: writing "*average*" followed by the label of the concept and the word *per*. Then, the label of the source concept of the structural relationship is added. Similarly, when the function name is *amount of*, *maximum* or *minimum*, the KPI is formed by adding the word *per* and the label of the source concept in the structural relationship. Lastly, in case of *number of* function, the KPI is formed by writing *number of* followed by the label of the source concept, in the structural relationship—by using its plural form.

Example

The application of the *leaf-attribute pattern* (pattern 1) is illustrated in Figure 6.

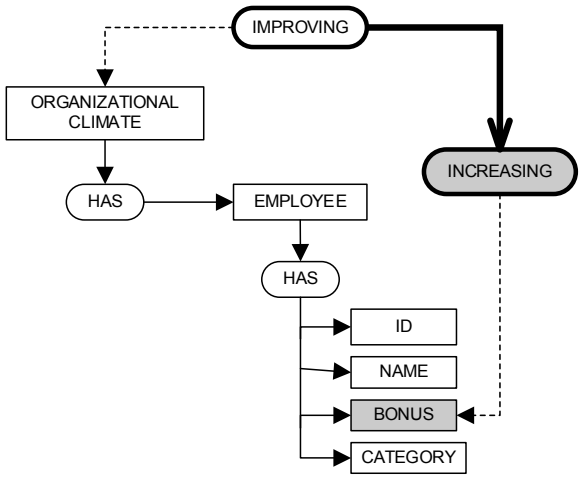


Fig. 6. Application of the leaf-attribute pattern. Source: The authors

A set of candidate KPIs is derived by following the process described by our proposal corresponding to pattern 1. As a result, the candidate KPIs are:

amount of BONUSES
maximum BONUS
minimum BONUS
amount of BONUSES per EMPLOYEE
maximum BONUSES per EMPLOYEE
minimum BONUSES per EMPLOYEE
average BONUSES
average BONUSES per EMPLOYEE
number of EMPLOYEES

2) The dynamic-relationship pattern.

Context

This pattern is applied when a *dynamic relationship* is a candidate for deriving a set of KPIs. An example of this structure is shown in Figure 7.

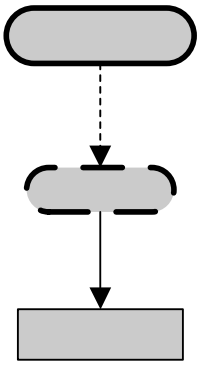


Fig. 7. Structure of dynamic-relationship pattern. Source: The authors

Problem

The problem relies on generating a set of KPIs from a candidate *dynamic relationship*, which is determined by using the pattern 2. Also, the type of *achievement relationship*—improvement, maintenance or accomplishment—linked to the concept should be identified. So, the appropriate KPI structure can be defined.

Structure

The elements involved in this pattern are as follows:

Concept. The target concept in the dynamic relationship.

Achievement relationship. The relation linked to the dynamic relationship. This relationship can be improvement-, maintenance-, or accomplishment-type.

Dynamic relationship. The relationship containing the action verb.

ISSN 2395-8618

Solution

In this paper, the *achievement relationships* involved in the case study are treated as *improvement*-type. More detailed information about *achievement-type relationships* is provided by Lezcano [16]. The process for applying this pattern is as follows.

A KPI is obtained by writing a function name from the list: *amount of* and *percentage of*. Then, the action verb is added by using its past participle. Lastly, the label of the target concept, in the dynamic relationship, is added.

Example

The application of the *dynamic-relationship pattern* is illustrated in Figure 8.

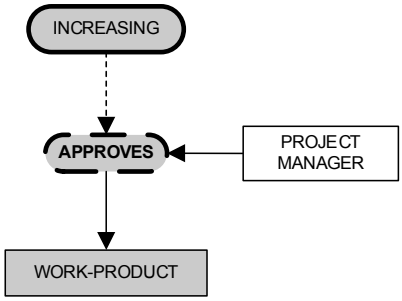


Fig. 8. Application of the dynamic-relationship pattern. Source: The authors

Figure 8 illustrates the occurrence of pattern 2, so a set of candidate KPIs is derived by following the process described in the solution Section corresponding to pattern 2. As a result, the candidate KPIs are:

amount of APPROVED WORK-PRODUCTS
percentage of APPROVED WORK-PRODUCTS

IV. Case Study

In this Section, we present a case study related to the organizational objective *IMPROVING SALES*. In this example, the domain related to such objective is modeled. So, *IMPROVING SALE* is represented by the achievement relationship IMPROVING and the class concept *SALE*. This organizational objective is in turn decomposed into two sub-objectives: *INCREASING REVENUES* and *INCREASING PRODUCT SELLING*.

After the relevant concepts and their relationships have been modeled, we can analyze the application of the proposed patterns. As a result, pattern 1 and pattern 2 are identified and a set of candidate KPIs are derived. This example is illustrated in Figure 9.

As a result, the set of candidate KPIs related to the organizational objective IMPROVING SALES are summarized in Table 2, including the pattern applied for selecting the candidate KPI.

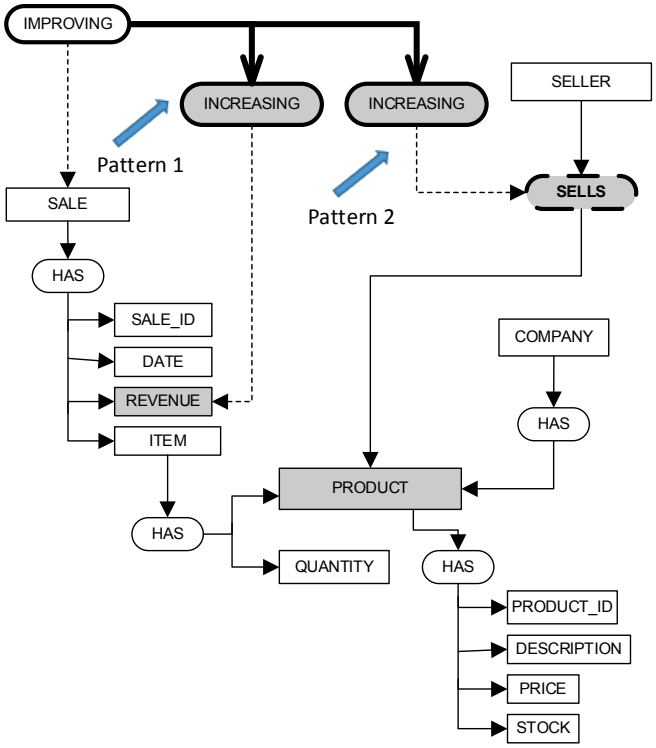


Figure 9. Application of the proposed method. Source: The authors.

TABLE II
Case Study About Candidate KPIs. Source: The authors

Candidate KPI	Pattern Applied
amount of REVENUES	1
maximum REVENUES	1
minimum REVENUES	1
average REVENUES per SALE	1
average REVENUES	1
amount of REVENUES per SALE	1
maximum REVENUES per SALE	1
minimum REVENUES per SALE	1
number of SALES	1
amount of SOLD PRODUCTS	2
percentage of SOLD PRODUCTS	2

V. Conclusions and Future Work

Strategic alignment involves holistic control of the organization. Such control should be assisted by appropriate methods that allow us to reason about the fulfillment of the organizational objectives. KPIs have demonstrated to be effective resources for assessing such fulfillment. By analyzing some contributions, we evidenced they exhibit some drawbacks for linking organizational objectives to KPIs. In this paper, we proposed a method based on patterns for deriving KPIs from organizational objectives as an appropriate solution for overcoming the identified drawbacks.

ISSN 2395-8618

Then, we demonstrated the application of the proposed method by using a lab study.

As a result, we experienced the advantages of using the pre-conceptual schemas as a mean for representing a particular domain. Such advantages rely on the proximity to the stakeholder language as well as the capacity for overcoming ambiguity problems. Moreover, the proposed patterns allowed for the derivation of candidate KPIs in a systematic way overcoming subjectivity problems. Thus, the use of such patterns demonstrated relevant advantages concerning the reuse, communication, and documentation of solutions to recurrent problems related to the generation of KPIs from goals.

Lastly, by applying the proposed method in this paper, the stakeholder is provided with a set of candidate KPIs intended to assess the fulfillment of the organizational objectives. Such KPIs are systematically obtained from a particular domain modeling, and they correspond with the own and specific needs an organization experiments. Since the candidate KPIs are systematically obtained, they can make aware the stakeholder about still undiscovered candidate KPIs.

Some lines of future work can be proposed: (i) the application of the proposed method on other different domains; (ii) the comparison of the results against the established indicators by specific frameworks; (iii) the full automation of the process for automatically deriving KPIs; (iv) the definition of new patterns for deriving candidate KPIs from goals; and (v) the automatic generation of source code from the candidate KPIs, to be included in the source code of the application to-be-developed.

REFERENCES

[1] M. Alyahya and M. A. Suhaimi, "A Conceptual Model for Business and Information Technology Strategic Alignment from the Perspective of Small and Medium Enterprises," *International Journal of Business, Humanities and Technology*, vol. 3, no. 7, pp. 83–90, 2013.

[2] C. Giannoulis, M. Petit, and J. Zdravkovic, "Modeling Business Strategy: A meta-model of Strategy Maps and Balance Scorecards," in *RCIS 2011: Research Challenges in Information Science*, France, 2013, pp. 1–6.

[3] A. Trendowicz, J. Heidrich, and K. Shintani, "Aligning Software Projects with Business Objectives," in *Joint Conference of the 21st International Workshop on Software Measurement and the 6th International Conference on Software Process and Product Measurement*, 2011.

[4] V. Basili, J. Heidrich, M. Lindvall, J. Münch, M. Regardie, D. Rombach, C. Seaman, and A. Trendowicz, "Linking Software Development and Business Strategy Through Measurement," *IEEE Computer*, vol. 43, no. 4, pp. 57-65, 2010.

[5] V. Basili, C. Lampasona, and A.E. Ocampo Ramírez, "Aligning Corporate and IT Goals and Strategies in the Oil and Gas Industry," in *PROFES 2013, LNCS 7983*, Springer-Verlag, 2013, pp. 184–198.

[6] F. Cosenz, "The Entrepreneurial University: A Preliminary Analysis Of The Main Managerial And Organisational Features Towards The Design Of Planning & Control Systems In European Academic Institutions," *Management Research and Practice*, vol. 5, no. 4, pp. 19-36, december, 2013.

[7] A. Kronz, "Managing of Process Key Performance Indicators as Part of the ARIS Methodology," *Corporate Performance Management: ARIS in Practice,* Springer-Verlag, pp. 31–44, 2006.

[8] HY. Wu, "Constructing a strategy map for banking institutions with key performance indicators of the balanced scorecard," *Evaluation and Program Planning,* vol. 35, pp.303–320, 2012.

[9] A. Shamsaei, A. Pourshahid, and D. Amyot, "Business Process Compliance Tracking Using Key Performance Indicators," *Lecture Notes in Business Information Processing*, pp. 73-84, 2011.

[10] A. Pourshahid, G. Richards, and D. Amyot, "Toward a goal-oriented, business intelligence decision-making framework," in *MCETECH 2011. LNBIP*, Heidelberg: Springer, 2011, vol. 78, pp. 100–115.

[11] D. Barone, L. Jiang, D. Amyot, and J. Mylopoulos, "Reasoning with Key Performance Indicators," *LNBIP*, ISSN: 1865-1348, vol. 92, pp. 82–96, 2011.

[12] A. Maté, J. Trujillo, and J. Mylopoulos, "Conceptualizing and Specifying Key Performance Indicators in Business Strategy Models," *LNCS*, ISSN: 0302-9743, pp. 282-291, 2012.

[13] S. Strecker, U. Frank, D. Heise, and H. Kattenstroth, "METRICM: a modeling method in support of the reflective design and use of performance measurement systems," *Inf Syst E-Bus Manage*, Springer Verlag, ISSN:1617-9854, pp. 241-276, 2012.

[14] U. Frank, D. Heise, H.Kattenstroth, and H. Schauer, "Designing and Utilising Business Indicator Systems within Enterprise Models – Outline of a Method," in *Modellierung Betrieblicher Informationssysteme (MobIS 2008)*, Germany, November 27-28, 2008.

[15] R. S. Kaplan, *Conceptual Foundations of the Balanced Scorecard*, Harvard Business School Accounting, 2010.

[16] L. A. Lezcano, "Elaboración Semiautomática de Diagrama de Objetivos," Ms.C. thesis, Universidad Nacional de Colombia, Colombia, 2007.

[17] V. Basili, A. Trendowicz, M. Kowalczyk, J. Heidrich, C. Seaman, J. Münch, and D. Rombach, "GQM+Strategies in a Nutshell. Aligning Organizations Through Measurement," *The Fraunhofer IESE Series on Software and Systems Engineering*, pp. 9-17, 2014. ISBN: 978-3-319-05046-1.

[18] L. Thevenet, "Modeling Strategic Alignment Using INSTAL," *Lecture Notes in Computer Science*, vol. 5232, pp. 261–271, 2011.

[19] K. Doumi, S. Baïna, and K. Baïna, "Modeling Approach Using Goal Modeling and Enterprise Architecture for Business IT Alignment," *Lecture Notes in Computer Science*, vol. 6918, pp. 249–261, 2011.

[20] C. Giannoulis, J. Zdravkovic, and M. Petit, "Model-Centric Strategy-IT Alignment:An Empirical Study in Progress," *Lecture Notes in Business Information Processing*, vol. 148, pp. 146-155, 2013.

[21] V. Popova and A. Sharpanskykh, "Modeling organizational performance indicators". Inf Syst. Elsevier. ISSN: 0306-4379, vol. 35 (4), pp. 505–527, 2010.

[22] C. M. Zapata, G. L. Giraldo, and S. Londoño, "Esquemas preconceptuales ejecutables," *Avances En Sistemas E Informática*, ISSN: 1657-7663 ed: Universidad Nacional De Colombia Sede Medellin, vol. 8, fasc.1, pp. 15-24, 2011.

[23] C. M. Zapata, A. Gelbukh, and F. Arango, "Pre-conceptual schema: A conceptual-graph-like knowledge representation for requirements elicitation," *Lecture Notes in Computer Science*, vol. 4293, pp. 17-27, 2006.

[24] E. Kavakli, "Goal-driven requirements engineering: Modeling and guidance," PhD thesis, University of Manchester, 1999.

[25] W. Engelsman, D. A. C. Quartel, H. Jonkers, and M. J. van Sinderen, "Extending enterprise architecture modelling with business goals and requirements," *Enterprise Information Systems*, vol. 5, pp. 9–36, 2011.

[26] E. Yu, M. Strohmaier, and X. Deng, "Exploring intentional modeling and analysis for enterprise architecture," in *Proceedings of the EDOC 2006 workshop on trends in enterprise architecture research (TEAR 2006)*, Hong Kong, Republic of China.

[27] J. L. De la Vara, J. Sánchez, and O. Pastor, "On the Use of Goal Models and Business Process Models for Elicitation of System Requirements,"

ISSN 2395-8618

Lecture Notes in Business Information Processing, vol. 147, pp. 168–183, 2013.

[28] G. Gröner, M. Asadi, B. Mohabbati, D. Gašević, F. Silva, and M. Bošković, "Validation of User Intentions in Process Models," *Lecture Notes in Computer Science*, vol. 7328, pp. 366–381, 2012.

[29] C. Giannoulis, M. Petit, and J. Zdravkovic, "Towards a Unified Business Strategy Language:A Meta-model of Strategy Maps," in *PoEM 2010, LNBIP,* 2010, vol. 68, pp. 205–216.

[30] R. S. Kaplan, and D. P. Norton, *Strategy Maps: Converting Intangible Assets into Tangible Outcomes*, Harvard Business School Publishing Corporation, 2004.

[31] I. Gmati, M. Missikoff, and S. Nurcan, "A Systematic Method for the Intentional Modeling and Verification of Business Applications," in *VII Conference of the Italian Chapter of AIS Information technology and Innovation trend in Organization*, Naples, Italy, 2010.

[32] E. Caputo, A. Corallo, E. Damiani, and G. Passiante, "KPI Modeling in MDA Perspective," *Lecture Notes in Computer Science*, pp. 384–393, 2010.

[33] A. Del-Río-Ortega, M. Resinas, C. Cabanillas, and A. Ruiz-cortéz, "On the definition and design-time analysis of process performance indicators," *Information Systems*, vol. 38, pp. 470–490, 2013.

[34] B. Wetzstein, Z. Ma, and F. Leymann, "Defining Process Performance Indicators : An Ontological Approach," *Lecture Notes in Business Information Processing*, pp. 227–238, 2008.

[35] C. Giannoulis, and J. Zdravkovic, "Modeling Strategy Maps and Balanced Scorecards using i*," in *iStar 2011*, Trento, Italy, 2011, pp. 90–95.

[36] A. Martinez, "Conceptual Schemas Generation from Organizational Models in an Automatic Software Production Process," Ph.D. thesis, Valencia University of Technology., Spain, and University of Trento, Italy, 2011.

ISSN 2395-8618

Optimizing Data Processing Service Compositions Using SLA's

Genoveva Vargas-Solar, José-Luis Zechinelli-Martini, and Javier-Alfonso Espinosa-Oviedo

***Abstract*—This paper proposes an approach for optimally accessing data by coordinating services according to Service Level Agreements (SLA) for answering queries. We assume that services produce spatio-temporal data through Application Programming Interfaces (API's). Services produce data periodically and in batch. Assuming that there is no full-fledged DBMS providing data management functions, query evaluation (continuous, recurrent or batch) is done through reliable service coordinations guided by SLAs. Service coordinations are optimized for reducing economic, energy and time costs.**

***Index Terms*—Data service, query optimization, workflow, service composition, SLA.**

I. Introduction

PERVASIVE denotes something "spreading throughout," thus a pervasive computing environment is the one that is spread throughout anytime anywhere and at any moment. From the computing science point of view what is interesting to analyze is how computing and software resources are available and provide services that can be accessed by different devices. For facilitating availability to these resources, they are wrapped under the same representation called service. A service is a resource handled by a provider, that exports an application programming interface (API). The API defines a set of method headers using an interface definition language. Consider a scenario where multiple users evolve within an urban area carrying GPS-enabled mobile devices that periodically transmit their location. For instance, the users location is notified by a stream data service with the following (simplified) interface:

```
subscribe() → {location:⟨nickname, coor⟩}
```

consisting of a subscription operation that, after invocation, will produce a stream of `location` tuples, each with a nickname that identifies the user and her coordinates. A stream is a continuous (and possibly infinite) sequence of tuples ordered in time.

The rest of the data is produced by the next two on-demand data services, each represented by a single operation:

```
profile(nickname) → {person:⟨age, gender,
                                email⟩}
interests(nickname) → {s_tag:⟨tag, score⟩}
```

The first provides a single `person` tuple denoting a profile of the user, once given a request represented by her nickname. The second produces, given the nickname as well, a list of `s_tag` tuples, each with a tag or keyword denoting a particular interest of the user (e.g. music, sports, fashion, etc.) and a score indicating the corresponding degree of interest.

Users access available services for answering some requirement expressed as a query. For instance, assume that Bob needs to find friends to make decisions whether he can meet somebody downtown to attend an art exposition. The query can be the following:

```
Find friends who are no more than
3 km away from me,
who are over 21 years old
and that are interested in art
```

But issuing the query from a mobile device, is not enough for evaluating it, some Service Level Agreements (SLA) need to also be expressed. For example, Bob wants the query to be executed as soon as possible, minimizing the battery consumption and preferring free data services.

Of course, the query cannot be solved by one service, some information will come for Google maps and Google location, other by Bob's personal directory, the availability of Bob's friend in their public agendas. Thus, the invocation to the different services must be coordinated by a program or script that will then be executed by an execution service that can be deployed locally on the user device or not. In order to do so, other key infrastructure services play an important role particularly for fulfilling SLA requirements. The communication service is maybe the most important one because it will make decisions on the data and invocation transmission strategies that will impact SLA.

Focussing on the infrastructure that makes it possible to execute the services coordination by making decisions on the best way to execute it according to given SLAs, we identify two main challenges:

– Enable the reliable coordination of services (infrastructure, platform and, data management) for answering queries.

Manuscript received on March 04, 2016, accepted for publication on June 15, 2016, published on June 25, 2016.

G. Vargas-Solar is Senior Scientist of the French Council of Scientific Research, LIG-LAFMIA Labs, France (web: http://www.vargas-solar.com).

J.L. Zechinelli Martini is a scientist at the LAFMIA Lab, France, – UDLAP Antena.

J.A. Espinosa-Oviedo is a scientist at the Barcelona Super Computer Centre, Spain, and a member of the LAFMIA lab, France.

ISSN 2395-8618

– Deliver request results in an inexpensive, reliable, and efficient manner despite the devices, resources availability and the volume of data transmitted and processed.

Research on query processing is still promising given the explosion of huge amounts of data largely distributed and produced by different means (sensors, devices, networks, analysis processes), and the requirements to query them to have the right information, at the right place, at the right moment. This challenge implies composing services available in dynamic environments and integrating this notion into query processing techniques. Existing techniques do not tackle at the same time classic, mobile and continuous queries by composing services that are (push/pull, static and nomad) data providers.

Our research addresses novel challenges on data/services querying that go beyond existing results for efficiently exploiting data stemming from many different sources in dynamic environments. Coupling together services, data and streams with query processing considering dynamic environments and SLA issues is an important challenge in the database community that is partially addressed by some works. Having studied the problem in a general perspective led to the identification of theoretical and technical problems and to important and original contributions described as follows. Section II describes the phases of hybrid query evaluation, highlighting an algorithm that we propose for generating query workflows that implement hybrid queries expressed in the language HSQL that we proposed. Section III describes the optimization of hybrid queries based on service level agreement (SLA) contracts. Section V discusses related work and puts in perspective our work with existing approaches. Section VI concludes the paper and discusses future work.

II. Hybrid Query Evaluation

We consider queries issued against data services, i.e., services that make it possible to access different kinds of data. Several kinds of useful information can be obtained by evaluating queries over data services. In turn, the evaluation of these queries depends on our ability to perform various data processing tasks. For example, data correlation (e.g. relate the profile of a user with his/her interests) or data filtering (e.g. select users above a certain age). We must also take into consideration restrictions on the data, such as temporality (e.g. users logged-in within the last 60 minutes).

We denote by "hybrid queries" our vision of queries over dynamic environments, i.e. queries that can be mobile, continuous and evaluated on top of push/pull static or nomad services. For example, in a mobile application scenario, a user, say Mike, may want to *Find friends who are no more than 3 km away from him, who are over 21 years old and that are interested in art*. This query involves three data services methods defined above. It is highly desirable that such query can be expressed formally through a declarative language named Hybrid Service Query Language (HSQL)[1]. With this goal in mind we adopt a SQL-like query language which is similar to CQL [1], to express the query as follows:

```
SELECT p.nickname, p.age, p.sex, p.email
FROM profile p, location l [range 10 min],
     interests i
WHERE p.age >= 21 AND
     l.nickname = p.nickname AND
     i.nickname = p.nickname AND
      'art' in i.s_tag.tag
      AND dist(l.coor, mycoor) <= 3000
```

The conditions in the `WHERE` clause enable to correlate `profile`, `location`, and `interests` of the users by their nickname, effectively specifying join operations between them. Additional conditions are specified to filter the data. Thus, users who are older than 21 and whose list of interests includes the tag `'art'`, and whose location lies within the specified limit are selected. For the location condition, we rely on a special function `dist` to evaluate the distance between two geographic points corresponding to the location of users, the current location of the user issuing the query is specified as `mycoor`. Since a list of scored tags is used to represent the interests of an user, we use a special `in` operator to determine if the tag `'art'` is contained in the list, while the list in question is accessed via a path expression.

Since the location of the users is subject to change and delivered as a continuous stream, it is neither feasible nor desirable to process all of the location data, therefore temporal constraints must be added. Consequently, the `location` stream in the `FROM` clause is bounded by a time-based window which will consider only the data received within the last 10 minutes. Given that the query is continuous, this result will be updated as the users' location changes and new data arrives. This is facilitated by a special `sign` attribute added to each tuple of the result stream, which denotes whether the tuple is added to the result (positive sign) or removed from it (negative sign).

In order to evaluate a declarative hybrid query like the one presented in the example we need to derive an executable representation of it. Such executable representation in our approach is a query workflow.

A. Query Workflow

A workflow fundamentally enforces a certain order among various activities as required to carry out a particular task. The activities required to evaluate a hybrid query fall into two basic categories: data access and data processing. Both of these types of activities are organized in a workflow following a logical order determined by the query. The execution of each of the activities, in turn, is supported by a corresponding service; data

[1]This language was proposed in the PhD dissertation of V. Cuevas Vicenttin of University of Grenoble.

ISSN 2395-8618

access activities by data services and data processing activities by computation services.

Following our service-based approach, the workflow used to evaluate a hybrid query consists of the parallel and sequential coordination of data and computation services. For example, the workflow representation for the query in the example is depicted in Figure 1.

The data services are represented by parallelograms, whereas computation services are represented as rounded rectangles and correspond to traditional query operators such as join or selection. The arrows indicating sequential composition not only imply order dependencies among the activities but also data dependencies; in particular, tuples that need to be transmitted between the different activities that produce and consume them.

Since the workflow enabling the evaluation of a given hybrid query acts in fact as a service coordination (comprising data and computation services), we refer to it as *query workflow*. Evaluating a hybrid query from a given query coordination depends first on finding the adequate (data and computation) services, second on their invocation, and finally on their communication and interoperation.

B. Computing a Query Workflow Cost

In order to use SLA's to guide query workflows evaluation, it is necessary to propose a cost model that can be used to evaluate a query workflow cost. The query workflow cost is given by a combination of QoS measures associated to the service methods it calls[2], infrastructure services such as the network and the hosting device it uses. The cost model considered for query workflows is defined by a combination of three costs: execution time, monetary cost and battery consumption. These costs are computed by calculating the cost of activities of a query workflow.

Query workflow cost: is computed by aggregating its activities costs based on its structure. The aggregation is done by following a systematic reduction of the query workflow such as in [2], [3]. For each sequential or parallel coordination, the reduction aggregates the activities costs. For the nested coordinations, the algorithm is applied recursively. The resulting cost is computed by a pondered average function of the three values.

$$\text{Cost} = \alpha\ (\text{temporal}_c) + \beta\ (\text{economic}_c) + \gamma\ (\text{energy}_c)\ /\ 3$$

We assume that the activity costs are estimated according to the way data are produced by the service (i.e., batch for on-demand services, continuous by continuous services).

Cost of an activity calling an on-demand service: For data produced in batch by on-demand services, the global activity cost is defined by the combination of three costs:

- *Temporal cost* given by (i) the speed of the network that yields transfer time consumption, determined by the data size and the network's conditions (i.e., latency and throughput) both for sending the invocation with its input data and receiving results; (ii) the execution of the invoked method has an associated approximated method response time which depends on the method throughput[3].
- *Economic cost* given by (i) the type of network: indeed, transmitting data can add a monetary cost (e.g., 3G cost for mega octets transfer); (ii) the cost of receiving results from a service method invocation sent to a specific service provider, for example getting the scheduled activities from the public agenda of my Friends can have a cost related to a subscription fee.
- *Energy cost* produced as a result of using the network and computing resources in the device hosting the service provider that will execute the method called. These operations consume battery entailing an energy cost.

Cost of an activity calling a continuous service cost: For data produced by continuous services the global activity cost is defined by the combination of three costs that depend on the data production rate resulting from the invocation of a method. The costs are multiplied by the number of times data must be pulled and transferred. The economic cost can be associated to a subscription model where the cost is determined by the production rate. For example receiving data frequently (e.g., give my current position every five minutes, where five minutes is the expected production rate) can be more expensive than receiving data in specific moments (e.g., the number of times Bob went to the supermarket during a month). The temporal and energy costs are also determined by the frequency in which data are processed (processing rate): data can be processed immediately, after a threshold defined by the number of tuples received, or the elapsed time, or a buffer capacity. Both production rate and processing rate impact the execution time cost, execution economic cost, and battery consumption cost.

III. Optimizing Hybrid Queries Using SLA Contracts

Given a hybrid query and a set of services that can be used for answering it, several query workflows can be used for implementing it. For example, consider the query workflow in Figure 2a that is a version of the friend finder example. The figure shows a query workflow coordinating activities in parallel for retrieving the profile and the location datasets. Then, the filtering activity that implements a window operator is placed just after the activity that retrieves the location. This activity reduces the input dataset size. Then, both datasets are correlated and finally the last activity filters the dataset to get data related only to 'Joe'. Placing the last activity just after of the retrieval of the profile dataset can reduce the processing time.

[2]QoS measures are a set of quantitative measures that describe the possible conditions in which a service method invocation is executed.

[3]The method throughput is given by the amount of requests in a period of time (e.g., each minute) and the state of the device such as memory or CPU.

ISSN 2395-8618

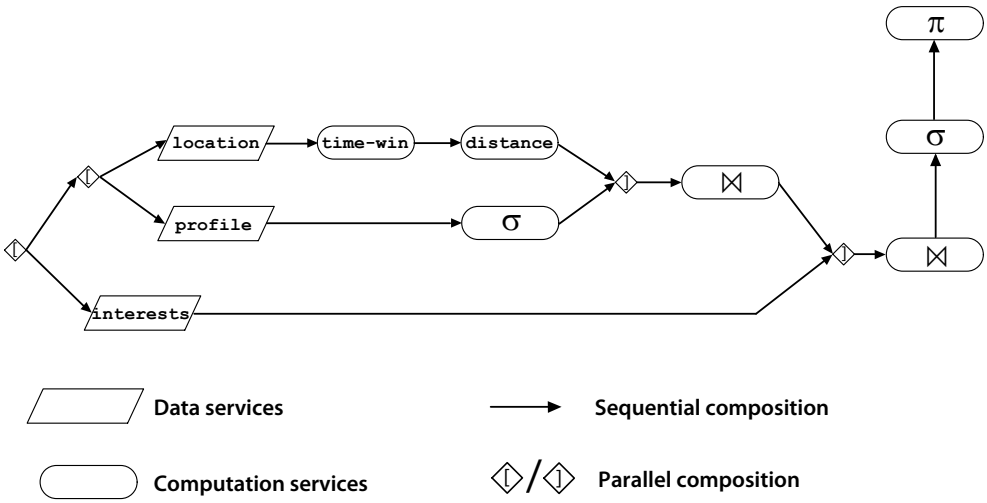


Fig. 1. Query coordination for the query the initial example

Now consider the query workflow in Figure 2b that coordinates activities sequentially. Each activity in the control flow consumes and produces data that at the end result in a dataset which is equivalent with the first one.

Optimizing a hybrid query implies choosing the query workflow that best implements it with respect to a given SLA. Similar to classic query optimization techniques, we propose an optimization process that consists in two phases: (i) generating its search space consisting in "all" the query workflows that implement a hybrid query and (ii) choosing the top-k query workflows with costs that are the closest to the SLA.

A. Generating Potential Query Workflow Space

We use rewriting operations (e.g. split, aggregate, parallelize, etc.) for generating a set of "semantically" equivalent query workflows The rewriting process is based on two notions: function and data dependency relationships.

- Function: represents a data processing operation. We consider the following functions:
  1) `fetch` for retrieving a dataset from a data provision service (e.g., get Bob's friends);
  2) `projection` of some of the attributes of each item (e.g., tuple) of a dataset (e.g., get name and location of Bob's friends assuming that there each friend has other attributes);
  3) `filter` the items of a dataset according to some criterion (e.g. Alice's friends located 3 Km from her current position); and,
  4) `correlation` of the items of two datasets according to some criterion (e.g., get friends shared by Bob and Alice that like "Art").
- Data dependency relationships between functions. Intuitively, given two functions with input parameters and an output of specific types, they are
  1) F_1 `independent` F_2 if they do not share input datasets;
  2) F_1 `concurrent` F_2 if they share common input datasets;
  3) F_1 `dependent` F_2 if they use common input datasets.

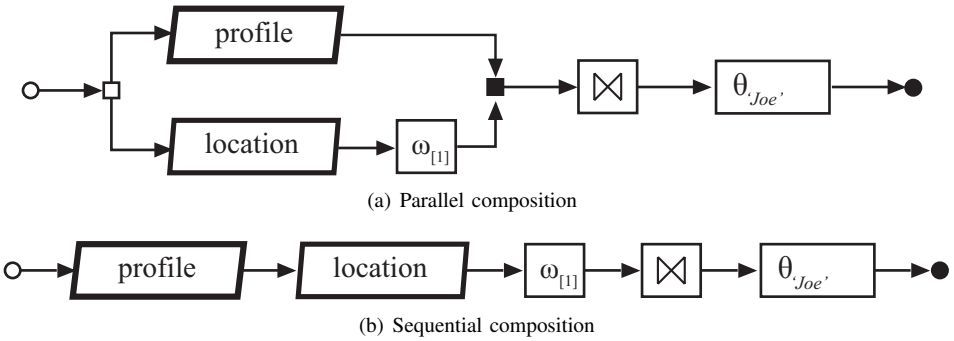
We propose rewriting rules and algorithms for generating a representation of an HSQL expression as a composite function and a data dependency tree. This intermediate representation is used for finally generating a query workflow search space. This generation is based on composition patterns that we propose for specifying how to compose the activities of a query workflow. Let F_1 and F_2 be functions of any type according to their dependency relationship they can give rise to two activities A_1 and A_2 related according to the composition patterns shown in Figure 3.

1) F_1 `independent` F_2 leads to three possible composition patterns: A_1 `sequence` A_2 or A_1 `sequence` A_2 or A_1 `parallel` A_2.
2) F_1 `concurrent` F_2 leads to the same sequential patterns of the previous case. In the case of the parallel pattern, it works only if and only if F_1 and F_2 are filtering functions or one of them is a filtering function and the other a correlation function.
3) F_1 `dependent` F_2 leads to a sequential composition pattern A_1 `sequence` A_2.

The search space generation algorithm ensures that the resulting query workflows are all deadlock free and that they terminate[4]. Once the search space has been generated, the query workflows are tagged with their associated three dimensional cost. Then, this space can be pruned in order to find the query workflows that best comply with the SLA

[4]Details on the algorithm can be found in the dissertation of Carlos Manuel López Enríquez of the University of Grenoble.

ISSN 2395-8618



(a) Parallel composition

(b) Sequential composition

Fig. 2. Query workflows of the *Search space*.

Relation	Notation	Composition pattern
A Independent of B	A||B	
A Concurrent with B	A⊳⊲B	
A Dependent on B	A⊳⊳B	

Fig. 3. Composition patterns

expressing the preferences of the user. This is done applying a top-k algorithm as discussed in the following section.

B. Computing an Optimization Objective

In order to determine which are the query workflows that answer the query respecting the SLA contract, we compute an optimization objective taking as input the SLA preferences and assuming that we know all the potential data and computing services that can be used for computing a query workflow. The SLA expressed as a combination of pondered measures, namely, execution time, monetary cost and energy. Therefore we propose an equation to compute a threshold value that represents the lowest cost that a given query can have given a set of services available and required for executing it and independently of the form of the query workflow (see Equation 1).

$$Opt(Q,R) = min(\sum_{j}(f(A_j,\Omega_j) - \gamma(Q,R_j))) \quad (1)$$

The objective is to find the combination of resources (R i.e., services) that satisfies a set of requirements (Q, i.e., the preferences expressed by the user and associated to a query). Every service participating in the execution of a query exports information about its available resources and used resources. For example the number of requests that a service can handle and the number of requests that are currently being processes. The principle of the strategy is described as follows: determine to which extent the required resources by Q can be fulfilled by a the resources provided by each service. The total result represents the combination of resources provided by available services that minimize the use of the global available resources (A) and the resources currently being used (Ω).

As shown in Figure 4 this value can be represented as a point in an n dimensional space where each dimension represents a SLA measure. Similarly, as shown in Figure 4, the query workflows cost which is also defined as a function of these dimensions, can be represented as a point in such n-dimensional space. The optimization process looks for points that are closest to the objective point by computing the Euclidean distance.

C. Choosing an Optimum Plan Family

We adopt a top-k algorithm in order to decide which of the k query workflows that represent the best alternative to implement a hybrid query for a given SLA. We adopt the Fagin's algorithm [4]. The top-k algorithm assumes m inverted lists $L_1, \ldots, L_m$ each of which is of the form $[\ldots, (qw_i, c_{i,j}), \ldots]$ with size $|\mathcal{S}|$ where $i \in [1 \ldots |\mathcal{S}|]$, and $j \in [i \ldots m]$. The order of the inverted lists depends on the algorithm.

The Fagin algorithm assumes that each list L_j is ordered by $c_{i,j}$ in ascending order. The principle is that the k query workflows are close to the top of the m lists. In the worst case, the $c_{k,j}$ is the last item for some $j \in [1, \ldots, m]$. The algorithm traverses in parallel the m lists by performing sequential access. Once k items have been visited in all the m lists, it performs random access over the m lists by looking for the already visited items and computes their scores. The scores are arranged in a sorted list in ascending order and thus the first k items are on the top of the score list. The main steps of the algorithm are:

1) Access in parallel the m lists by performing sequential access.
2) Stop once k query workflows have been seen in the m lists.

ISSN 2395-8618

$$QWF^1(p_1^1, p_2^1, \ldots, p_m^1)$$
$$QWF^2(p_1^2, p_2^2, \ldots, p_m^2)$$
$$\vdots$$
$$QWF^n(p_1^n, p_2^n, \ldots, p_m^n)$$

$$\forall i : 1 \leq i \leq n$$
$$d(p^i, o) = \sqrt{\sum_{j=1}^{m} (p_j^i - o_j)}$$

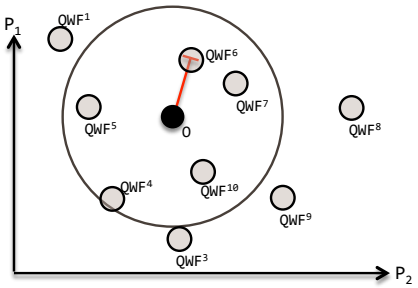


Fig. 4. Optimization approach

3) Perform random access over the m lists to obtain the m scaled attributes of each query workflow that has been processed and compute its $score$.
4) Sort in ascending order the scores.
5) Return the k query workflows on the top.

It is applied for obtaining an ordered family of query workflows that compose the optimum plan family. According to a descending order, the first query workflow will be executed. The rest of the queries can be stored as knowledge and they can be used for further optimizations. We do not consider learning based optimization [5] but we believe that such a technique can be applied in this case.

IV. Implementation and Experiments

We developed a proof of concept of our approach by implementing a service-based hybrid query processor named Hypatia for enacting query workflows. Figure 5 left side presents its architecture. The system is based on the Java platform. Queries in Hypatia are entered via a GUI and specified in our HSQL query language. Once a query is provided to the system it is parsed and then its corresponding query workflow is generated according to the algorithm that we described in Section III-A. The query parser and the query workflow constructor components perform these tasks. The parser was developed using the ANTLR[5] parser generator. The GUI also enables the user to visualize the query workflow, as well as the sub-workflows corresponding to composite computation services, which is facilitated by the use of the JGraph[6] library.

To implement stream data services we developed a special-purpose stream server framework, which can be extended to create stream data services from sources ranging from text files to devices and network sources. This framework employs Web Service standards to create subscription services for the streams. Stream data access operators in query workflows subscribe to these services and also receive the stream via special gateway services.

The evaluation of a query is enabled by two main components that support the computation services corresponding to data processing operations. A scheduler determines which service is executed at a given time according to a predefined policy. Composite computation services communicate via asynchronous queues and are executed by an ASM interpreter that implements our workflow model.

[5]http://www.antlr.org/
[6]http://www.jgraph.com/

A. Hybird Query Optimizer

We implemented a HYbrid Query OptimiZer (HyQoZ) that we integrated to the hybrid query evaluator Hypatia. Figure 5 shows the component diagram for processing hybrid queries with HyQoZ. An application representing a data consumer expresses hybrid queries based on the information about service instances provided by the APIDirectory, and define the SLA to fulfill. Applications may require either the evaluation of the hybrid query or the optimum query wrkflow implementing the hybrid query for its further execution. In such cases applications request either the evaluator Hypatiaor the optimizer HyQoZ.

Its components are described by REST interfaces and they exchange self-descriptive messages. The messages instantiate different coordinations for implementing the hybrid queries optimization. The interfaces and messages turn our optimization approach self-contained.

– Hypatiaaccepts the hybrid query evaluation requests. HybridQP validates the expression according to the information provided by the APIDirectory. QEPBuilder derives the optimization objective from the SLA contract and requests the hybrid query optimization to HyQoZ. The resulting query workflow is executed by the QWExecutor.
– HyQoZaccepts hybrid query optimization requests and looks for the satisfaction of the optimization objectives derived from the SLA contracts. HyQoZis composed by a series of components that implement the optimization stages.

Internally, HyQoZis composed by a series of orthogonal components that together perform the optimization. Components exchange self-descriptive messages carrying the required information for articulating the optimization.

ISSN 2395-8618

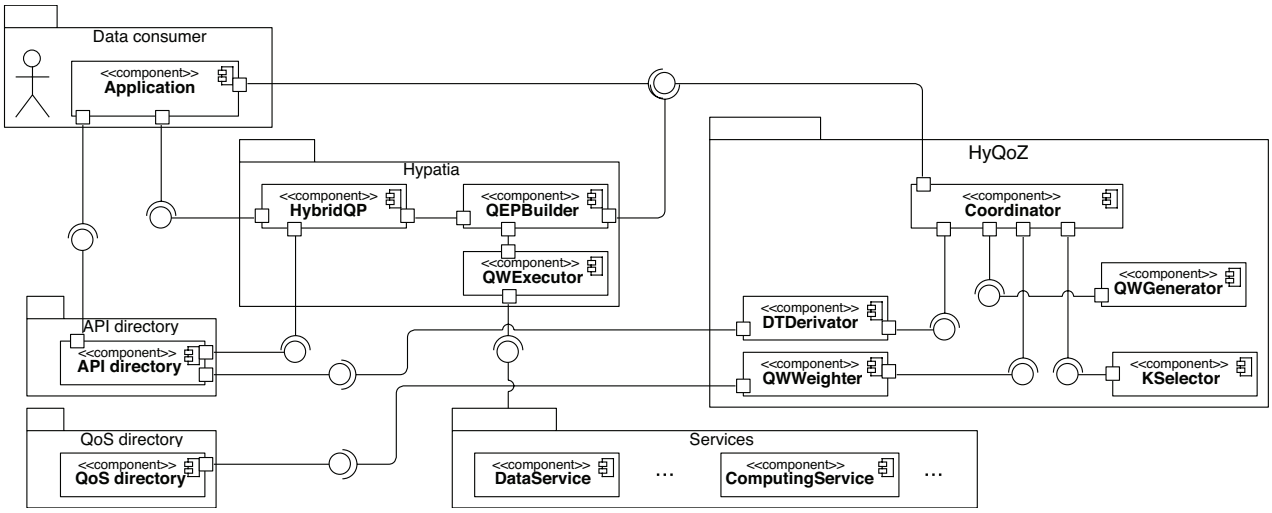


Fig. 5. Hybrid query processing components

B. Validation

We implemented two test scenarios and their corresponding data services to validate our approach. The first one, mainly a demonstration scenario, is the location-based application. In order to implement the Friend Finder scenario described in the Introduction we developed a test dataset using GPS tracks obtained from everytrail.com. Concretely, we downloaded 58 tracks corresponding to travel (either by walking, cycling, or driving) within the city of Paris. We converted the data from GPX to JSON and integrated it to our stream server to create the location service. For the profile and interests services we created a MySQL database accessible via JAX-WS Web Services running on Tomcat. The profile data is artificial and the interests were assigned and scored randomly using the most popular tags used in Flickr and Amazon. For the nearest-neighbor (NN) points of interest we converted a KML file[7] containing the major tourist destinations in Paris into JSON, this data is employed by the corresponding NN service in conjunction with the R-tree spatial indexation service. Finally, we implemented an interface based on Google Maps that enables to visualize the query result, which is presented in Figure 6.

The second scenario was developed to measure the efficiency of our current implementation in a more precise manner; it is based on the NEXMark benchmark[8]. Our main goal was to measure the overhead of using services, so we measured the total latency (i.e. the total time required to process a tuple present in the result) for a set of six queries (see Table I); first for our service-based system and then for an equivalent system that had at its disposal the same functionality offered by the computation services, but supported by objects inside the same Java Virtual Machine.

[7]Keyhole Markup Language https://developers.google.com/kml/documentation/

[8]http://datalab.cs.pdx.edu/niagara/NEXMark/

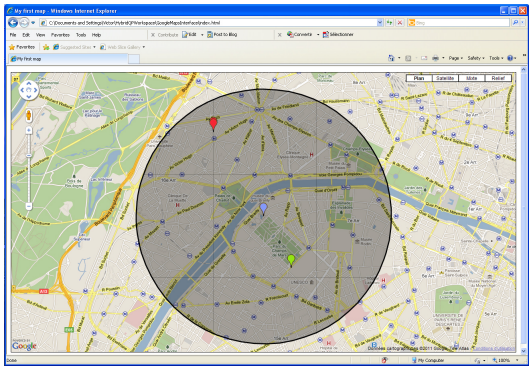


Fig. 6. Friend Finder visualization GUI

NEXMark proposes an auctions scenario consisting of three stream data services, **`person`**, **`auction`** and, **`bid`**, that export the following interfaces:

`person:`$\langle$`person_id`^{f}`, name`^{f}`, phone`^{f}`, email`^{f}`, income`$^f\rangle$

`auction:`$\langle$`open_auction_id`^{f}`, seller_person`^{f}`, category`^{f}`, quantity`$^f\rangle$

`bid:`$\langle$`person_ref`^{f}`, open_auction_id`^{f}`, bid`$^f\rangle$

Auctions work as follows. People can propose and bid for products. Auctions and bids are produced continuously. Table I shows the six queries that we evaluated in our experiment; they are stated in our HSQL language and for each we provide the associated query workflow and equivalent operator expression that implements them (generated by HYPATIA). Queries $\mathbf{Q}_1$ – $\mathbf{Q}_2$ mainly exploit temporal filtering using window operators, filtering and correlation with and/split-join like control flows. $\mathbf{Q}_3$ involves grouping and aggregation functions. $\mathbf{Q}_4$ adds a service call to a sequence of data processing activities with filtering and projection operations. Finally, $\mathbf{Q}_5$ – $\mathbf{Q}_6$ address several correlations organized in and/split-join control flows.

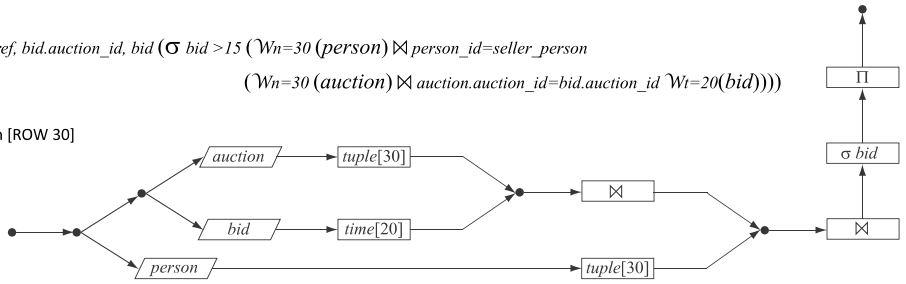
TABLE I
NEXMARK QUERIES

For the last 30 persons and 30 products offered, retrieve the bids of the last 20 seconds greater than 15 euros

1)

```
SELECT bids.person_ref, bid.auction_id, bid.bid
FROM bid [RANGE 20], auction [ROW 30], person [ROW 30]
WHERE bid.auction_id = auction.auction_id AND
auction.seller_person = person.person_id
AND bid.bid > 15;
```

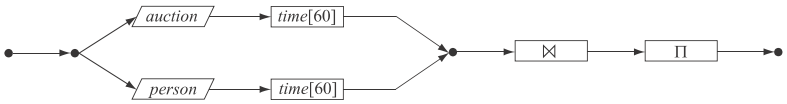
π *person_ref, bid.auction_id, bid* (σ *bid >15* ($\mathcal{W}$*n=30* (*person*) $\bowtie$ *person_id=seller_person*
($\mathcal{W}$*n=30* (*auction*) $\bowtie$ *auction.auction_id=bid.auction_id* $\mathcal{W}$*t=20*(*bid*))))



For the persons joining and the products offered during the last minute, generate the name and email of the person along with the id of the product he/she offers

2)

```
SELECT P.name, P.email, A.auction_id
FROM auction [RANGE 60] as A, person [RANGE 60] as P
WHERE A.seller_person = P.person_id;
```

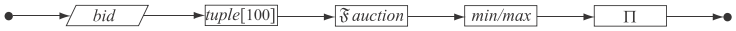


π *name, email, auction_id* ($\mathcal{W}$*t=60* (*person*) $\bowtie$ *person_id=seller_person* $\mathcal{W}$*t=60* (*auction*))

For the last 100 bids, find the maximum and minimum bid for each product

3)

```
SELECT bid.person_ref, bid.auction_id,
bid.bid, max(bid.bid), min(bid.bid)
FROM bid [ROWS 100]
GROUP BY bid.auction_id;
```

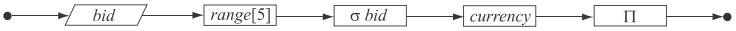


π *person_ref, auction_id, bid, max(bid), min(bid)* ($\mathfrak{F}$ *auction_id* ($\mathcal{W}$*t=60* (*auction*)))

Among the bids made in the last 5 seconds, find those whose amount is between 50 and 100 euros and their dollar equivalent

4)

```
SELECT bid.person_ref, bid.auction_id, bid.bid,
currency('EUR', 'USD', bid.bid)
FROM bid [RANGE 5]
WHERE bid.bid > 50.0 and bid.bid < 100.0;
```



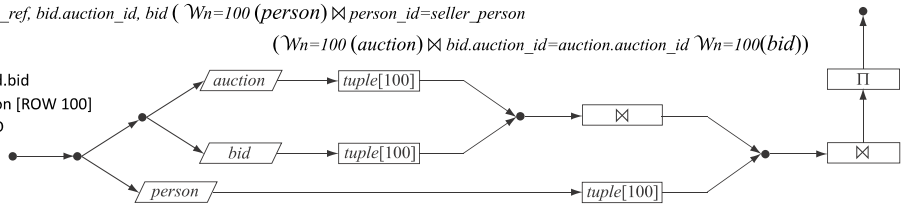
π *person_ref, auction_id, bid, currency('EUR', 'USD', bid)* (σ *bid >50 and bid <100* ($\mathcal{W}$*t=5* (*bid*)))

For the last 100 persons, products and bids; give the id of the seller person, the id of the product, and the amount of the bid

5)

```
SELECT bid.person_ref, bid.open_auction_id, bid.bid
FROM bid [ROW 100], auction [ROW 100], person [ROW 100]
WHERE bid.auction_id = auction.auction_id AND
auction.seller_person = person.person_id;
```

person_ref, bid.auction_id, bid ($\mathcal{W}$*n=100* (*person*) $\bowtie$ *person_id=seller_person*
($\mathcal{W}$*n=100* (*auction*) $\bowtie$ *bid.auction_id=auction.auction_id* $\mathcal{W}$*n=100*(*bid*))

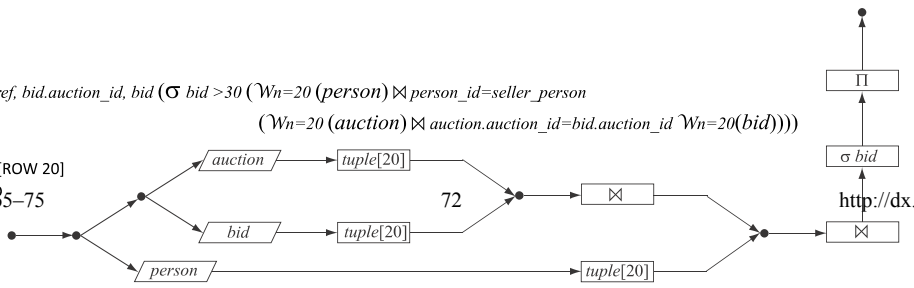


For the last 20 persons, products and bids; give the id of the seller person, the id of the product, and the amount of the bid, whenever that amount is greater than 30

6)

```
SELECT bid.person_ref, bid.auction_id, bid.bid
FROM bid [ROW 20], auction [ROW 20], person [ROW 20]
WHERE bid.auction_id = auction.auction_id AND
auction.seller_person = person.person_id
AND bid.bid > 30;
```

person_ref, bid.auction_id, bid (σ *bid >30* ($\mathcal{W}$*n=20* (*person*) $\bowtie$ *person_id=seller_person*
($\mathcal{W}$*n=20* (*auction*) $\bowtie$ *auction.auction_id=bid.auction_id* $\mathcal{W}$*n=20*(*bid*))))



 http://dx.doi.org/10.17562/PB-

ISSN 2395-8618

C. Experimental Results

For our experiments we used as a local machine a Dell D830 laptop with an Intel Core 2 Duo (2.10 GHz) processor and equipped with 2 GB of RAM. We also employed as a remote machine a Dell Desktop PC with a Pentium 4 (1.8 GHz) processor and 1 GB of RAM. In both cases running JSE 1.6.0_17, the local machine under Windows XP SP3 and the remote under Windows Server 2008.

As said before, to validate our approach we established a testbed of six queries based on our adaptation of the NEXMark benchmark. These queries include operators such as time and tuple based windows as well as joins. We measured tuple latency, i.e. the time elapsed from the arrival of a tuple to the instant it becomes part of the result, for three different settings. The first setting corresponds to a query processor using the same functionality of our computation services, but as plain java objects in the same virtual machine. In the second we used our computation services, which are based on the JAX-WS reference implementation, by making them run on a Tomcat container in the same machine as the query processor. For the third setting we ran the Tomcat container with the computation services on a different machine connected via intranet to the machine running the query processor.

The results are shown in Figure 7, and from them we derive two main conclusions. First, the use of services instead of shared memory resulted in about twice the latency. Second, the main overhead is due to the middleware and not to the network connection, since the results for the local Tomcat container and the remote Tomcat container are very similar. We believe that in this case the network costs are balanced-out by resource contingency on the query processor machine, when that machine also runs the container. We consider the overhead to be important but not invalidating for our approach, especially since in some cases we may be obliged to use services to acquire the required functionality.

From our experimental validation we learned that it is possible to implement query evaluation entirely relying on services without necessarily using a full-fledged DBMS or a DSMS (Data Stream Management Systems). Thereby, hybrid queries that retrieve on demand and stream data are processed by the same evaluator using well adapted operators according to their characteristics given our composition approach. The approach can seem costly because of the absence of a single DBMS, the use of a message based approach for implementing the workflow execution, and because there is no extensive optimization in the current version of HYPATIA. Now that we have a successful implementation of our approach, we can address performance issues further in order to reduce cost and overhead.

For validating HYQOZ, we developed the testbed that

1) generates synthetic hybrid queries,
2) generates the search space of query workflows following a data flow or control flow, and
3) estimates either the cost by means of a simulation using synthetic data statistics.

We used precision and recall measures to determine the proportion of interesting query workflows that are provided by the optimizer. The precision and recall are around 70% and 60% respectively.

V. RELATED WORK

In dynamic environments, query processing has to be continuously executed as services collect periodically new information (e.g., traffic service providing information at given intervals about the current state of the road) and the execution context may change (e.g., variability in the connection). Query processing should take into account not only new data events but also data providers (services) which may change from one location to another.

Existing techniques for handling continuous spatio-temporal queries in location-aware environments (e.g., see [6]–[11]) focus on developing specific high-level algorithms that use traditional database servers [12]. Most existing query processing techniques focus on solving special cases of continuous spatio-temporal queries: some like [8], [10], [11], [13] are valid only for moving queries on stationary objects, others like [14], [15] (Carney et al. 2002) are valid only for stationary range queries. A challenging perspective is to provide a complete approach that integrates flexibility into the existing continuous, stream, snapshot, spatio-temporal queries for accessing data in pervasive environments.

The emergence of data services has introduced a new interest in dealing with these “new” providers for expressing and evaluating queries. Languages as Pig and LinQ combine declarative expressions with imperative ones for programming queries, where data can be provided by services. In general, query rewriting, optimization and execution are the evaluation phases that need to be revisited when data are provided by services and they participate in queries that are executed in dynamic environments. Query rewriting must take into consideration the data service interfaces, since some data may need to be supplied to these in order to retrieve the rest of the data. The existence of a large number of heterogeneous data services may also necessitate the use of data integration techniques. In addition, new types of queries will require the definition of new query operators.

Traditional query optimization techniques are not applicable in this new setting, since the statistics used in cost models are generally not available. Furthermore, resources will be dynamically allocated via computation services, rather than being fixed and easy to monitor and control. Finally, query execution must also be reconsidered. First, the means to access the data is via services rather than scanning or employing index structures. Second, to process the data we depend on computation services, instead of a rigid DBMS.

The work [16] proposes a service coordination approach, where coordinations can optimized by ordering the service

ISSN 2395-8618

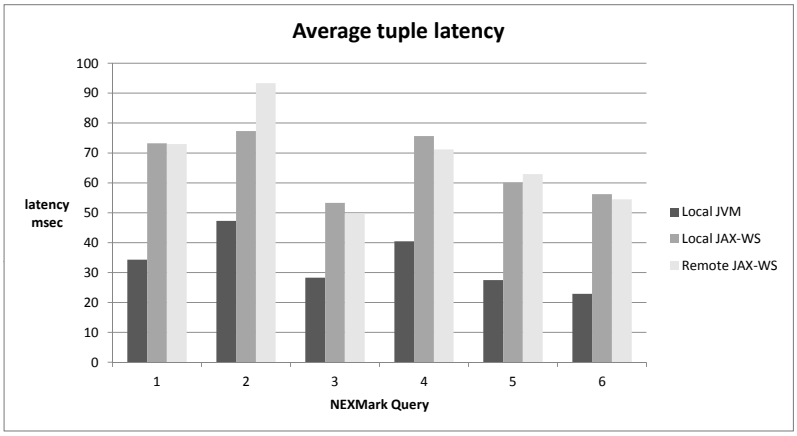


Fig. 7. Tuple latency for a services-based vs. a single Java application query processor

calls in a pipelined fashion and by tuning the data size. The control over data size (i.e., data chunks) and selectivity statistics are key assumptions adopted by the approach. Another aspect to consider during the optimization is the selection of services, which can have an impact on the service coordination cost. The authors of [3], [17] optimize service coordinations by proposing a strategy to select services according to multidimensional cost. Service selection is done by solving a multi objective assignment problem given a set of abstract services defined by the coordination. Services implementing the coordination can change but the control flow of the coordination remains the same.

The emergence of the map-reduce model, has introduced again parallelization techniques. Queries expressed in languages such as Pig[9], and SCOPE [18] can be translated into map/reduce [19] workflows that can be optimized. The optimization is done by intra-operator parallelization of map/reduce tasks. The work [20] applies safe transformations to workflows for factorizing the map/reduce functions, partitioning of data, and reconfiguring functions. Transformations hold preconditions and postconditions associated to the functions in order to keep the data flow consistency. The functional programming model PACT [21] extends the map/reduce model to add expressiveness that are black boxes within a workflow. In [22] the black boxes are analyzed at build-time to get properties and to apply conservative reorderings to enhance the run-time cost. The map/reduce workflows satisfy the need to process large-scale data efficiently w.r.t. execution time. Although we do not address query optimization under such context in the present work, we provide a discussion of its related issues and possible solutions in [23].

VI. Conclusion and Future Work

This paper presented our approach for optimizing service coordinations implementing queries over data produced by data services either on-demand or continuously. Such queries are implemented by query workflows that coordinate data and computing services. The execution of query workflows has to respect Service Level Agreement contracts that define an optimization objective described by a vector of weighted cost attributes such as the price, the time, the energy. The weights define the preferences among the cost attributes for enabling the comparison among query workflows. Our approach for generating the search space of query workflows that can optimize service coordinations, the cost estimation, and the solution space are oriented to satisfy SLA contracts.

[9]http://pig.apache.org

Acknowledgment

The authors would like to thank Víctor Cuevas Vicenttin and Carlos Manuel López Enríquez who were at the origin of the work presented here. We also thank the ANR ARPEGE project OPTIMACS that financed part of the work.

References

[1] A. Arasu, S. Babu, and J. Widom, "The CQL continuous query language: semantic foundations and query execution," *The VLDB Journal*, vol. 15, no. 2, pp. 121–142, 2006.

[2] U. Jaeger, "SMILE — a framework for lossless situation detection," in *In Proc. Int'l. Workshop on Inf. Technologies and Sys*, 1995, pp. 110–119.

[3] H. Wada, P. Champrasert, J. Suzuki, and K. Oba, "Multiobjective optimization of SLA-aware service composition," *2008 IEEE Congress on Services - Part I*, pp. 368–375, Jul. 2008.

[4] R. Fagin, A. Lotem, and M. Naor, "Optimal aggregation algorithms for middleware," *Journal of Computer and System Sciences*, vol. 66, no. 4, pp. 614–656, Jun. 2003.

[5] L. Martínez, C. Collet, C. Bobineau, and E. Dublé, "The QOL approach for optimizing distributed queries without complete knowledge," in *IDEAS*, 2012, pp. 91–99.

[6] R. Benetis, S. Jensen, G. Kariauskas, and S. Saltenis, "Nearest and reverse nearest neighbor queries for moving objects," *The VLDB Journal*, vol. 15, no. 3, pp. 229–249, Sep. 2006. [Online]. Available: http://dx.doi.org/10.1007/s00778-005-0166-4

[7] I. Lazaridis, K. Porkaew, and S. Mehrotra, "Dynamic queries over mobile objects," in *Proceedings of the 8th International Conference on Extending Database Technology: Advances in Database Technology*, ser. EDBT'02. London, UK, UK: Springer-Verlag, 2002, pp. 269–286. [Online]. Available: http://dl.{ACM}.org/citation.cfm?id=645340.650229

ISSN 2395-8618

[8] Z. Song and N. Roussopoulos, "K-Nearest neighbor search for moving query point," in *Proceedings of the 7th International Symposium on Advances in Spatial and Temporal Databases*, ser. SSTD'01. London, UK, UK: Springer-Verlag, 2001, pp. 79–96. [Online]. Available: http://dl.{ACM}.org/citation.cfm?id=647227.719093

[9] Y. Tao, X. Xiao, and R. Cheng, "Range search on multidimensional uncertain data," *ACM Trans. Database Syst.*, vol. 32, no. 3, Aug. 2007. [Online]. Available: http://doi.{ACM}.org/10.1145/1272743.1272745

[10] J. Zhang, M. Zhu, D. Papadias, Y. Tao, and D. L. Lee, "Location-based spatial queries," in *Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, ser. SIGMOD'03. New York, NY, USA: ACM, 2003, pp. 443–454. [Online]. Available: http://doi.{ACM}.org/10.1145/872757.872812

[11] B. Zheng and D. L. Lee, "Semantic caching in location-dependent query processing," in *Proceedings of the 7th International Symposium on Advances in Spatial and Temporal Databases*, ser. SSTD'01. London, UK, UK: Springer-Verlag, 2001, pp. 97–116. [Online]. Available: http://dl.{ACM}.org/citation.cfm?id=647227.719097

[12] M. F. Mokbel, X. Xiong, and W. G. Aref, "SINA: Scalable incremental processing of continuous queries in spatio-temporal databases," in *Proceedings of the ACM SIGMOD International Conference on Management of Data, Paris, France, June 13-18, 2004*, G. Weikum, A. C. Koenig, and S. Dessloch, Eds. ACM, 2004, pp. 623–634.

[13] O. Wolfson, A. P. Sistla, S. Chamberlain, and Y. Yesha, "Updating and querying databases that track mobile units," *Distrib. Parallel Databases*, vol. 7, no. 3, pp. 257–387, Jul. 1999. [Online]. Available: http://dx.doi.org/10.1023/A:1008782710752

[14] M. Hadjieleftheriou, G. Kollios, D. Gunopulos, and V. J. Tsotras, "On-line discovery of dense areas in spatio-temporal databases," in *SSTD*, 2003, pp. 306–324.

[15] S. Prabhakar, Y. Xia, D. V. Kalashnikov, W. G. Aref, and S. E. Hambrusch, "Query indexing and velocity constrained indexing: Scalable techniques for continuous queries on moving objects," *IEEE Trans. Comput.*, vol. 51, no. 10, pp. 1124–1140, Oct. 2002. [Online]. Available: http://dx.doi.org/10.1109/TC.2002.1039840

[16] U. Srivastava, K. Munagala, J. Widom, and R. Motwani, "Query optimization over web services," *VLDB'06, Proceedings of the 32nd International Conference on Very Large Data Bases*, 2006.

[17] D. Claro Barreiro, P. Albers, and J.-k. Hao, "Selecting web services for optimal composition," in *International Conference on Web Services (ICWS05)*, 2005.

[18] R. Chaiken, B. Jenkins, and Larson, "Scope: easy and efficient parallel processing of massive data sets," *VLDB*, 2008.

[19] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters," *Communications of the ACM*, vol. 51, no. 1, pp. 1–13, 2008.

[20] H. Lim, "Stubby: A transformation-based optimizer for MapReduce workflows," *VLDB*, vol. 5, no. 11, pp. 1196–1207, 2012.

[21] D. Battré, S. Ewen, F. Hueske, and O. Kao, "Nephele/Pacts: A programming model and execution framework for web-scale analytical processing," *Proceedings of the 1st ldots*, 2010.

[22] F. Hueske, M. Peters, M. J. Sax, and A. Rheinl, "Opening the black boxes in data flow optimization," *VLDB*, vol. 5, no. 11, pp. 1256–1267, 2012.

[23] V. Cuevas-Vicenttin, G. Vargas-Solar, C. Collet, and P. Bucciol, "Efficiently coordinating services for querying data in dynamic environments," *Mexican International Conference on Computer Science*, vol. 0, pp. 95–106, 2009.

ISSN 2395-8618

Recommendation for an Enterprise Content Management (ECM) Based on Ontological Models

José Márquez, Manuel Escalante, Leonardo Sampedro,
Elba Sánchez, Laura Ortiz, and Eduardo Zurek

***Abstract*—This paper presents a system of recommendations for an enterprise content manager (ECM) based on ontological models. In many occasions the results of a search are not accurate enough, so the user of the ECM system must check them and discard those not related to the search. In order to make recommendations, a proposal where it is necessary to review the instances of the ontological model is presented to manage the alias and ambiguities. Comparisons are made between the results obtained from the traditional search model and the recommendations suggested by the model proposed in this work.**

***Index Terms*—Ontologies, ECM, natural language processing, searching, recommendations, semantic web.**

I. Introduction

IN an Enterprise Content Management (ECM) system, the search is a critical and repetitive task. Access to the requested information is vital for the person who performs the search, but the information is not always presented explicitly, as when the search is done by date and author. For this reason, the person must read the documents and determine if the result is correct or not. In the market, there are different commercial solutions that implement ontological models in an ECM, such as Athento ECM [1] – Zaizi [2], but they do not reveal how to use ontological models because this is a commercial secret. For a content management user, it is important to have all the documents organized and have all the control access for the documents.

This paper describes recommendation system based on ontological models. The models give solution to two of the most common problems: ambiguity and alias, which are handle in order to give the final user some suggestions about other documents that could have any relation with the search terms.

The work described here is part of a research project founded by COLCIENCIAS, the entire project is aimed to the development of a recommendation system for an ECM software.

Two ontological models were applied to represent entities from the content of the documents. The results of applying the FOAF [3] model, with the property TheSameAs, can be used to present to the final user documents that are related with some person but are referenced with a nickname or alias, and cannot be reached with the traditional model of search. The second model has a special property, HasFacet, which enable the instances of the model to have relations with instances of other models such as those that we show here with car [4] and places [5] model.

This paper has seven parts. Section 1 introduces the work presented here. Section 2 describes some issues with the current search technique based on key words. Section 3 shows works that has been done by some ECM companies, and research on using semantic, ontology, and ECM to manage data and information in a different way. Section 4 presents information about ontology. Section 5 describes our proposal to handle ambiguity and alias problems on ECM. In section 6 we present some results after applying our proposal to a search engine, and show the differences with the current search method, and finally in Section 7 we present some conclusion of this work.

II. Problem

The problem we address can be formulated as follows: "Enterprise Content Management (ECM) makes reference to the strategies, methods and tools to capture, manage, storage, preserve and present the contents handled by an organization" [6].

In an ECM, it is not possible to find more relations between the objects that are part of the system, but only those established in the database design. Basically, in an ECM, we can make consultations about documents in a specific status, to consult the name, date or any other metadata, or find a word

Manuscript received on February 02, 2016, accepted for publication on May 24, 2016, published on June 25, 2016.

J. Marquez, E. Zurek, and M. Escalante are with the Universidad del Norte, Km. 5 Autopista a Puerto Colombia, Barranquilla, Colombia (e-mail:{jmarquez, ezurek}@uninorte.edu.co, wolverineun@hotmail.com).

L. Sampedro, E. Sánchez, and L. Ortiz are graduate students in Systems Engineerning, Universidad del Norte, Km. 5 Autopista a Puerto Colombia, Barranquilla, Colombia (e-mail: {ljsampedro, bita.sanchez2991, lauraortizmartinez}@gmail.com).

ISSN 2395-8618

in the main index, if it is indexed. To discover additional information like documents from people who are not users of the system, or documents where these people play an important role, they can be modeled following ontologies.

To make modifications that allow us to find new relations among the objects that are part of the system is not a simple task if it is made from the DB. For this reason, the use of ontologies is proposed to create models that define new relations and provide more information in the system.

In an ECM, documents are handled as such. Documents are understood as any form to present information, no matter its format or content. The ECM can manage resumes, contracts, invoices, mails, PQR, brochures, recipes, reports, researches, etc. It does not matter either its digital format (word, Excel, PDF).

Considering this variety, the ECMs choose to create metadata common to all, modeling them as a document. For this, metadata are created following some schemes that allow them to organize in hierarchy the information and save related information with characteristics of the document (physical location, format, entry date in the system).

In a search engine based on key words as ECMs commonly do, generally there are failures when alias or pseudonyms and names changes are handled. Another problem arises when dates not handled by the metadata are searched and they are in different formats. For instance, we will not get the same results from the date “01/23/2013” as we look for “Twenty Third Day of January 2013”, even if they make reference to the same date.

With the creation of an ontological model, basic relations of hierarchy can be established, as well as more elaborated relations that allow to associate objects of different classes [7].

III. Related works

The semantic web is a group of techniques and technologies to represent the knowledge in a specific domain [8]. These techniques allow sharing and reusing the information among applications and communities. Technologies such as RDF (Resource Description Framework)[9] and OWL (Ontology Web Language) to represent knowledge [10] are used by companies to create software like Athento [1] and Zaizi [2], which add an improvement to the search engine of the ECM system.

The traditional search engines base their operation in the use of inverted indexes, which enable a high speed of response [11]. With the use of semantic, and indexing documents with relevant search terms (relevant to each document), allow Athento users to navigate through documents that are related.

In the works currently under development, it is always attempted to apply or develop an ontological model to represent the domain managed by the ECM like in the OpenCalais project [12]. In this work, it is pretended to handle the ambiguities and alias that could be present in the text of the document.

Some authors propose the use of ontology in ECM software to manage the problem of ambiguous representation of knowledge with two approaches, ex post and ex ante. Ex post try to solve the ambiguity once the information has been collected, on the other hand ex ante try to avoid the ambiguity before it happens [13].

The use of ontology models can help to build a structure that represents the possible class that a document could belong to, and can be used to classified documents in a repository. Instances of the ontology model can be use to tag the content too, and could be used to let the final user choose the document type that he or she like the most [14].

Some collaborative work require the interchange of recorded data to accomplished a job, most of the time is hard to share with collaborators or to reuse the data (or information) in some other process, because the data is not in the correct format or can has different meaning from one data base to other. Some authors propose to solve this problem by using an information system in conjunction with ontology models to give an easy way to access the information [15]. In this approach the use of ontology can help to build a semantic tag system to accurately annotate the document from repository, and give the final user the ability to access the knowledge present on the ontological relations [15]. The main objective of this approach is to let the final user spend less time on preprocessing the data for exchange with coworker, or reuse the data in some other different process.

Semantic and ontology models are used to give a web page a more structure and search engine friendly form. The W3C has proposed some tags that help the web programmer to build more structured web page. For example, it is possible to use the tags <article> reference, article, and comments about that article. By using this tag is possible to share and reuse data with applications, enterprises, and communities [16].

It is possible to treat the ambiguity with the creation of classes in the ontological model and declaring them as disjointed. For example, with the word blackberry, which can make reference to a cell phone brand or a fruit, the classes shown in Figure 1 could be created.

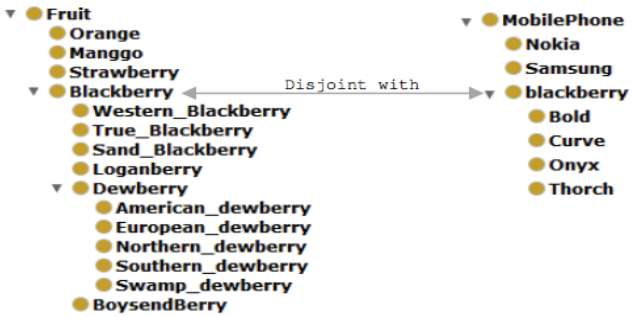


Fig. 1. Disjoint Class example.

ISSN 2395-8618

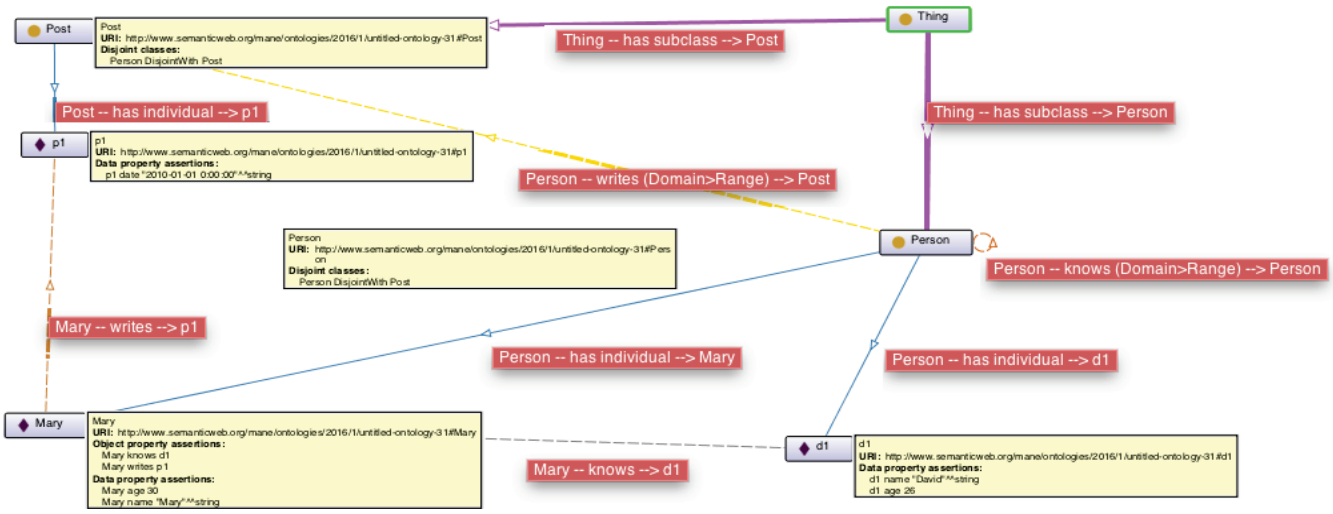


Fig. 2. Example of ontology model with Protégé

IV. Ontology

Ontologies and semantics have become a very important subject in the last years, which are researched by different academic groups. Ontologies are used in software design to establish communication among actors of design, interfaces and communications design, and knowledge discovering [17].

To create an ontology model we can follow the same steps as Object Oriented software design, to create a formal specification of the terms that belongs to a domain. To represent all this information we can use classes, attributes, relations, and instances. The relations let us express how objects from the domain are related with objects of the range.

We can build hierarchy with classes and sub classes, and explicit express what classes are disjoint, for example on figure 2 we have Person class and Post class, that are disjoint and are related by the relations "writes"; this relation express that a person writes a post. The dotted lines express relation with objects of the same class, for example "knows" shows the relation between persons.

The studies for the use of ontologies are made in order to apply them in a legal environment [18], considering as a base the definitions (content, intellectual property, instantiation) provided by the Dublin Core framework [19], in order to have information of the information, but the idea is not only focused on using the hierarchy developed with the model, but also to create relations that can provide more knowledge [18].

Ontologies are also used to share knowledge among systems, to allow the communication among intelligent agents, and in the software development to identify requirements and set tasks [20].

V. Proposed Approach

The use of ontological models is proposed to manage the problems previously mentioned. For this purpose, it is necessary to create instances of the ontological models with the information of the entities presented in the text of each document. In order to control or handle the alias, the use of the relation "theSameAs" and the relation "hasFacet" to handle the ambiguities is proposed. These relations must be integrated in the ontological model and used at the moment of the creation of instances. Once instances are created, they are indexed in order to be found quickly at the moment of the search. All this is based on the communication between the ECM (using the communication CMIS standard of the ECMs) and a module to create ontological instances.

In an ECM, the nature of documents can be varied and depends on the use given by the company that uses them. We can find recipes, invoices, resumes and articles. The use of a unique model would not be good, because it could leave out entities that represent important information. As we can observe in Figure 3, a model to represent people, institutions and publications with the relation "theSameAs", is used.

A good basis for the ontological models is FOAF [3], which represents the relations of people (friendOf, fatherOf, etc.) and their basic data (name, last name, etc.), then the system can be enriched with models such as resumeRDF [21] that represents the information of the resume and organization [8], which gathers the information about the organizations that could be related to a person.

A. Alias handle

The problem of the alias arises when a document inside of the ECM uses an alias to make reference to an entity in the system. The alias is identified as an instance in the model, and a relation "theSameAs" is created between the alias and the instance referred to.

To make the search the following steps must be followed:

ISSN 2395-8618

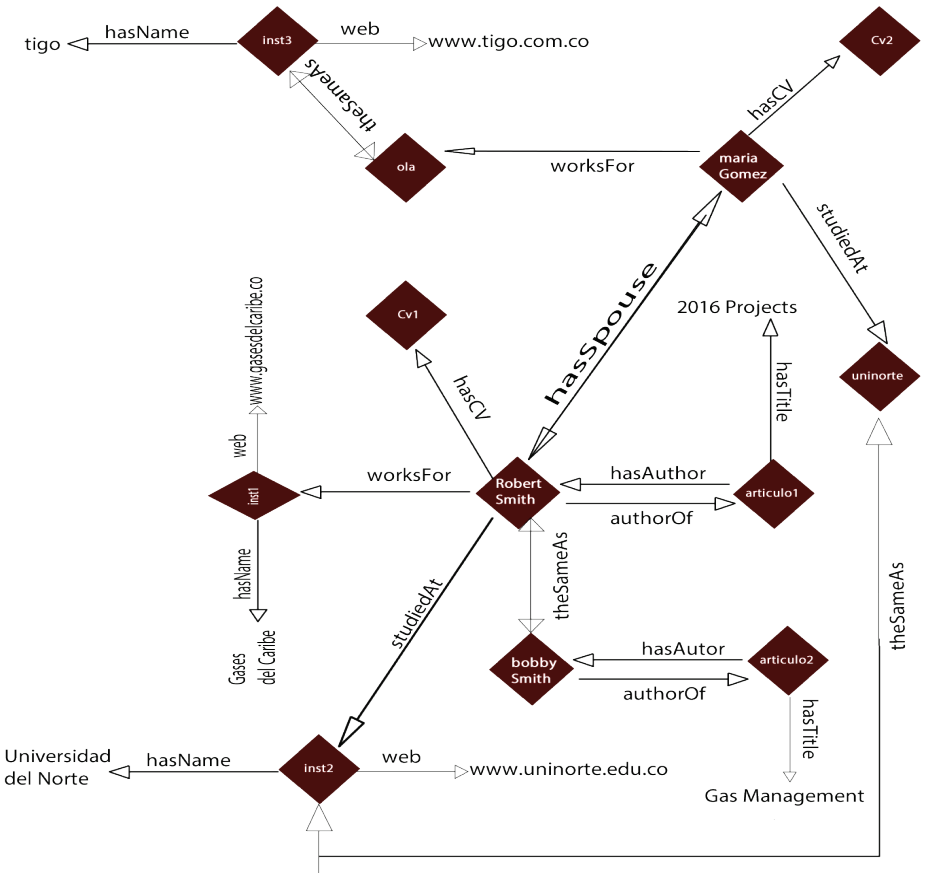


Fig. 3. Instance of ontological model with alias.

1. Indexes (where the texts of each document are indexed) are consulted, and a list of results with the found coincidences is created.
2. Indexes of the ontological models instances are checked:
   - 2.1 Instances whose name matches with the searched words are consulted, and their relations with other individuals are added to a list.
   - 2.2 The individual obtained from the previous step is consulted using "theSameAs". Then the relations and properties of most interest (authorOf) are consulted and added to the list.
3. The obtained results are organized in two lists and presented to the user.

In Figure 3, "RobertSmith" and "BobbySmith" entities make reference to the same person. If the user makes a search with the words "Robert Smith", the relation "theSameAs" between the two entities allows recommending the document "Gas management".

B. Handling ambiguity

Ambiguity arises when the user searches a word that can have more than one meaning, and the search engine shows as result any document that contains the searched word, without taking in consideration the meaning of the word in the text [22]. With the relation "hasFacet" in an ontological model, this situation can be handled and represent the different meanings that a word can have.

In order to show to the final user the different meanings that a word can have, these steps must be followed:

1. Indexes (where the text of each document is indexed) are consulted, and a list of results with the found coincidences is created.
2. Indexes of the ontological models instances are checked:
   - 2.1 Instances whose name matches with the searched words are consulted, and their relations with other individuals are added to a list.
   - 2.2 The individual obtained from the previous step is consulted using "hasFacet". Then the relations

ISSN 2395-8618

and properties of most interest (authorOf) are consulted and added to the list.

3. The obtained results are organized in two lists and presented to the user.

The goal is to have a recommendation system based on ontologies for an ECM, but these steps here described could be used in any search engine after making the necessary changes.

In the tests, Abox ECM [23] has been used, a web application ran under Win7, Sqlserver [24] and .Net framework 4.5 [24]. All the code for the handling of ontologies and the instances was developed with C# [25] and the library dotNetRDF [26]. The recommendations system was developed following the design pattern MVC in order to be shown inside of the application Abox [23]. A controller that makes the process previously described was developed, which communicates with the ECM by means of the CMIS standard (Figure 4). A view was also created, and whose principal task is to create a block (Figure 5) with the recommendations. The work and handling of the ontological models were developed with the tool Protégé [27, 28].

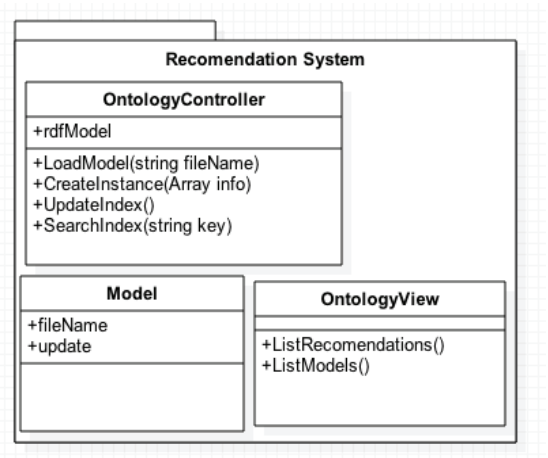


Fig. 4. Diagram of the system.

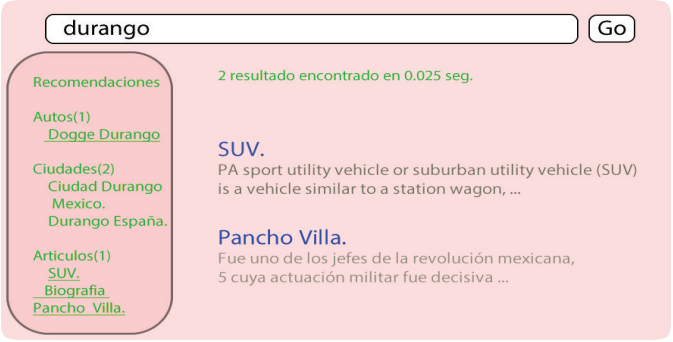


Fig. 5. Result view with <DIV> block(on the left) that have the recommendations.

VI. Results

All the tests were conducted with the ECM Abox [23], a repository of 4322 documents, the FOAF models [3], organization, and place [29].

In the repository, there are some documents with ambiguities, and non-related documents, but in the text, there is the word "Durango", which makes reference to a place in Mexico, a place in Spain, and a car model.

To handle this case, the "Durango" entity was created, and the relation "hasFacet" was used for each of the different instances referred to.

Using the instances of the Figure 6, it is possible to suggest the user all the possible meanings of the word "Durango" inside of the system.

In Figure 5, it is presented the HTML view of the Abox searcher in which the list of recommendations is embedded in. It can be observed that the results obtained with the traditional system (right) are not clear for the user. The system has found a total of 15 documents that have to be read by the user to discard those that are not related to the search. In the left side, the list of recommendations created by the proposed system can be observed and which presents to the user the different aspects registered in the system regarding the searched word. With this list, the user can discard as quickly as possible (they do not have to read the unnecessary documents) the documents not related to its search.

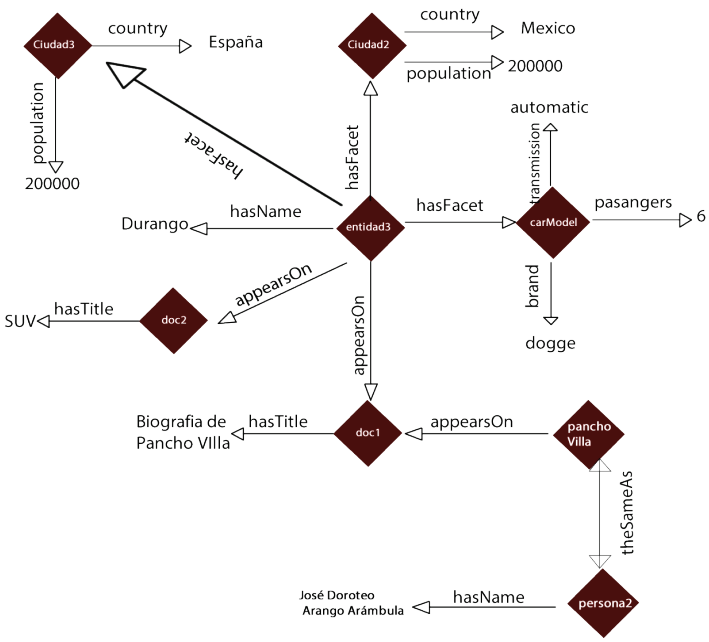


Fig. 6. Instance of ontological model with ambiguity.

VII. Conclusions

Showing the relation among entities that appear in the text of a document is an advantage for the ECM user.

ISSN 2395-8618

In this development, model FOAF [3] is used because it has many relations, but the steps here described can be used with any ontological model, and it is even recommended to use multiple models with the purpose of representing the largest amount of entities that have a relevant meaning in the business logic of the system.

Providing suggestions to the user helps him to make decisions which documents are relevant for its search and save time. In this work, ambiguities are handled with a relation in the ontological model, but it is also possible to do it by defining a class and a subclass for every meaning that a word can have, which is not practical because it should previously be known which are the ambiguous terms and their different meanings.

We show in this publication the advantages of the use of ontologies, not only to represent metadata, but also to represent the entities present in the text of the document.

ACKNOWLEDGMENTS

This work was supported by the Departamento Administrativo de Ciencia, Tecnología e Innovación (COLCIENCIAS, Colombia) through 562-2012 Banco de Proyectos y Programas I+D+I – Co-funding Modality, which funds the Project 121556236053.

REFERENCES

[1] "Software ECM | Athento." [Online]. Available: http://www.athento.com/software-enterprise-content-management/. [Accessed: 07-Jul-2015].

[2] "Home | Zaizi." [Online]. Available: http://www.zaizi.com/. [Accessed: 07-Jul-2015].

[3] "The FOAF Project." [Online]. Available: http://www.foaf-project.org/. [Accessed: 07-Jul-2015].

[4] "Car sales Ontology" [Online]. Avilable: http://www.heppnetz.de/ontologies/vso/ns . [Accesed: 15- Jan - 2015]

[5] "GeonNames Ontology" [Online], Avilable: http://www.geonames.org/ontology/documentation.html . [Accessed: 10-Jan-2015]

[6] "AIIM - The Global Community of Information Professionals." [Online]. Available: http://www.aiim.org/. [Accessed: 29-Apr-2015].

[7] G. Barchini, M. Álvarez, and S. Herrera, "Information systems: new ontology-based scenarios," *JISTEM-J. Inf. Syst. Technol. Manag.*, vol. 3, no. 1, pp. 2–18, 2006.

[8] "World Wide Web Consortium (W3C)." [Online]. Available: http://www.w3.org/. [Accessed: 07-Jul-2015].

[9] "RDF - Semantic Web Standards." [Online]. Available: http://www.w3.org/RDF/. [Accessed: 29-Apr-2015].

[10] G. Antoniou and F. Van Harmelen, *A semantic web primer*. MIT press, 2004.

[11] S. Brin and L. Page, "Reprint of: The anatomy of a large-scale hypertextual web search engine," *Comput. Netw.*, vol. 56, no. 18, pp. 3825–3833, 2012.

[12] "Thomson Reuters | Open Calais." [Online]. Available: http://new.opencalais.com/. [Accessed: 07-Jul-2015].

[13] Jan Von Brocke, "Enterprise content management in information system research, foundations, methods and cases", Springer, 2014.

[14] Daniela Briola et al. 2013, "Ontologies in industrial Enterprise Content Management Systems: the EC2M Project".

[15] Abdelkader Hameurlain et al. "Transaction on Large-Scale Data- and knowledge-centered systems IV". Springer, 2011.

[16] W3C semantic elements. [Online]. Available: http://www.w3schools.com/html/html5_semantic_elements.asp . [Accessed: 9-Jul-2015].

[17] G. N. Aranda and F. Ruiz, "Clasificación y ejemplos del uso de ontologías en Ingeniería del Software," in *XI Congreso Argentino de Ciencias de la Computación*, 2005.

[18] D. Tiscornia, "The LOIS project: Lexical ontologies for legal information sharing," in *Proceedings of the V Legislative XML Workshop*, 2006, pp. 189–204.

[19] "DCMI Home: Dublin Core® Metadata Initiative (DCMI)." [Online]. Available: http://dublincore.org/. [Accessed: 07-Jul-2015].

[20] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowl. Acquis.*, vol. 5, no. 2, pp. 199–220, 1993.

[21] "ResumeRDF Ontology Specification." [Online]. Available: http://rdfs.org/resume-rdf/. [Accessed: 07-Jul-2015].

[22] J. Lyons, *Linguistic semantics: An introduction*. Cambridge University Press, 1995.

[23] "ECM Abox | Adapting." [Online]. Available: http://www.adapting.com/index.php/abox-ecm/. [Accessed: 07-Jul-2015].

[24] "Microsoft – Official Home Page." [Online]. Available: https://www.microsoft.com/en-gulf/. [Accessed: 07-Jul-2015].

[25] "Visual C#." [Online]. Available: https://msdn.microsoft.com/en-us/library/kx37x362.aspx. [Accessed: 29-Apr-2015].

[26] "dotNetRDF - Semantic Web, RDF and SPARQL Library for C#/.Net." [Online]. Available: http://www.dotnetrdf.org/. [Accessed: 07-Jul-2015].

[27] N. F. Noy and D. L. McGuinness, "Desarrollo de Ontologías-101: Guía para crear tu primera ontología," 2005.

[28] "Protégé." [Online]. Available: http://protege.stanford.edu/. [Accessed: 29-Apr-2015].

[29] "Organization - schema.org." [Online]. Available: https://schema.org/Organization. [Accessed: 07-Jul-2015].

ISSN 2395-8618

A Memetic Algorithm Applied to the Optimal Design of a Planar Mechanism for Trajectory Tracking

Eduardo Vega-Alvarado, Edgar Alfredo Portilla-Flores, Efrén Mezura-Montes,
Leticia Flores-Pulido, Maria Bárbara Calva-Yáñez

***Abstract*—Memetic algorithms (MA), explored in recent literature, are hybrid metaheuristics formed by the synergistic combination of a population-based global search technique with one or more local search algorithms, which in turn can be exact or stochastic methods. Different versions of MAs have been developed, and although their use was focused originally on combinatorial optimization, nowadays there are memetic developments to solve a wide selection of numerical type problems: with or without constraints, mono or multi objective, static or dynamic, among others. This paper presents the design and application of a novel memetic algorithm, MemMABC, tested in a case study for optimizing the synthesis of a four-bar mechanism that follows a specific linear trajectory. The proposed method is based on the MABC algorithm as a global searcher, with the addition of a modified Random Walk as a local searcher. MABC is a modified version of the Artificial Bee Colony algorithm, adapted to handle design constraints by implementing the feasibility rules of Deb. Four-bar mechanisms are a good example of hard optimization problems, since they are used in a wide variety of industrial applications; simulation results show a high-precision control of the proposed trajectory for the designed mechanism, thus demonstrating that MemMABC can be applied successfully as a tool for solving real-world optimization cases.**

***Index Terms*—Four-bar mechanism, hard optimization, memetic algorithm, random walk.**

I. Introduction

METAHEURISTICS are algorithms designed to solve a wide variety of hard optimization problems in an approximate way, using trial and error techniques. The general characteristics of a metaheuristic are: it is inspired on natural or artificial processes, uses stochastic components (involving random variables), and has a series of parameters that must be adjusted to the specific case [1].

Manuscript received on March 06, 2016, accepted for publication on June 20, 2016, published on June 25, 2016.

Eduardo Vega-Alvarado is with Universidad Autónoma de Tlaxcala, Facultad de Ciencias Básicas, Ingeniería y Tecnología and with Instituto Politécnico Nacional, Centro de Innovación y Desarrollo Tecnológico en Cómputo, Mexico (e-mail: evega@ipn.mx).

Leticia Flores-Pulido is with Universidad Autónoma de Tlaxcala, Facultad de Ciencias Básicas, Ingeniería y Tecnología, Mexico (e-mail: aicitel.flores@gmail.com).

Edgar Alfredo Portilla-Flores and Maria Bárbara Calva-Yáñez are with Instituto Politécnico Nacional, Centro de Innovación y Desarrollo Tecnológico en Cómputo, Mexico (e-mail: aportilla@ipn.mx, b_calva@hotmail.com).

Efrén Mezura Montes is with Universidad Veracruzana, Centro de Investigación en Inteligencia Artificial, Mexico (e-mail: emezura@uv.mx).

Population-based metaheuristics start from an initial set of solutions or proposed individuals to find the optimal value or values of a problem, and there are two general groups of these algorithms: evolutive computing and swarm intelligence. The two main tasks of modern metaheuristics are diversification (exploration) and intensification (exploitation) [2]. Techniques based on population are good for exploring, but usually are deficient when exploiting [1]. Several alternatives have been developed in order to solve this weakness, and Memetic Algorithms (MAs) highlight between them for synergically combining the global search dynamics of a population metaheuristic with the refinements of a local search (LS), in order to obtain a hybrid method of solution [3], [4].

The first MAs were taken with suspicion because of their metaheuristic nature, but in the last decade these techniques have been applied successfully to solve different problems including numerical cases with constraints, and dynamic multi-objective optimization [3], [5]–[7]. In the real world, they have been used in security and cryptanalysis [8], control systems [9], task scheduling and routing [10], and data classification [11], among others.

In mechanical engineering, *synthesis* is the design process of machines or mechanical systems [12]. The purpose of the mechanism determines the type of synthesis to carry out: generation of motion, function or trajectory. In the synthesis for function generation, an input motion to the mechanism is correlated with another for output; in trajectory generation, a point is controlled to track a line in a plane, such that it assumes a prescribed set of sequential positions [13]. This work addresses the dimensional synthesis of a mechanism: calculate the length of the necessary links for tracking a specific trajectory.

The four-bar mechanism has been widely used in engineering design, since it is the simplest articulated mechanism for controlled movement with a degree of freedom. The synthesis of four-bar mechanisms for trajectory tracking is a well-known numerical problem previously explored in depth, and classical approaches have been used for this synthesis, including graphical and analytical methods; however, they have a limitation regarding the number of points to be tracked, since solutions are extremely complicated for problems with more than four points. For this reason, the design of these

ISSN 2395-8618

mechanisms is a typical case of hard numerical optimization.

Hard optimization problems can't be solved in an optimal way or to a guaranteed limit using deterministic methods and with normal computing resources. Taking into account that most real-world optimization cases are hard problems, it is necessary to develop alternative methods for their solution. In the search of new ideas for improving the performance of metaheuristics several models have been implemented, based on natural or artificial processes.

In this work, the design of a new memetic algorithm (MemMABC) is presented, with its application to the optimal synthesis of a four-bar mechanism for the control of a trajectory delimited by a set of N precision points tracked by the coupler. MemMABC is a combination of MABC and a version of the Random Walk algorithm (RW) modified for handling design constraints by implementing the feasibility rules of Deb. Although these building blocks are well-known methods usually they are applied only in an individual way; this is a novel approach to solve constrained problems of numerical optimization by combining synergically two strategies of different nature, applied to a real-world engineering case.

The paper is organized as follows: Section 2 shows the basic model of memetic algorithms and the implementation of local search stages. In Section 3, the MemMABC algorithm is introduced, with an analysis of both its global and local searchers. Section 4 describes the problem of mechanism synthesis, with a brief explanation of the kinematics. A case study with a specific optimization problem is analyzed in Section 5, including the description of the design variables. Section 6 presents the applied algorithm, with special emphasis on its computational implementation. Finally, experimental results are reported in Section 7, while the conclusions of this paper are included in Section 8.

II. Memetic Algorithms

Early work on MAs dates from the 80s [1], [14]; as a consequence of the strengthening in evolutive algorithms, new ideas were implemented in order to improve their performance, once that their limitations were known. In 1989, Moscato [15] proposed the memetic algorithms, to simulate the cultural evolution process derived from Lamarck's evolution theory and the *meme*, presented by R. Dawkins as the equivalent to the gene in natural evolution. MAs constitute a general method based on the sinergetic combination of algorithms for global and local search, in a new optimization philosophy [16]. A meme corresponds to recurrent patterns in real world or to specific knowledge, and is coded for the effective solution of problems as the building block of cultural know-how that is transmissible and reproducible [17], [18].

Three stages can be identified in the evolution of MAs [4]:

1) Applications are made by simple combination of an evolutive algorithm and a specialized method of local search.
2) Multiple memes are used, and the developments include any population-based algorithm as a global search tool.
3) Explicit mechanisms of learning are incorporated (adaptive MAs), with the use of exact methods in tandem configuration for local search.

MAs of first and second stages are still being developed because of their simplicity and efficiency, with an improvement over the performance of previously applied single metaheuristics.

A. Basic Model of a Memetic Algorithm

Figure 1 shows the block diagram of a basic population metaheuristic, indicating the four points where a local search can be included in order to form a MA [19], [20]:

1) On the population, to simulate the cultural development that will be transmitted from one generation to another; it can be applied to the whole set of agents or to specific elements, and even to the initial group.
2) On the parent or selected parents, before reproduction stage.
3) When new solutions are generated, to produce a better offspring.
4) On the offspring, before selecting a survivor according to fitness criteria.

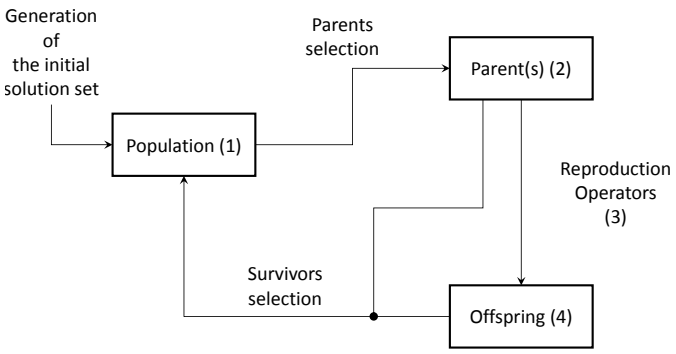


Fig. 1. Block diagram of a basic population metaheuristic

From this basic model several versions of MAs have been developed, differing between each other in at least one of the following aspects:

– Population metaheuristic used as a base.
– Selected algorithm for local search (exact method or metaheuristic, number of memes to consider.)
– Conditions for local search (trigger event, frequency, intensity, number of individuals to improve, etc.) [18], [21].

Originally MAs were based on evolutive algorithms but nowadays their implementation with swarm intelligence methods is common, using algorithms such as Particle Swarm Optimization (PSO) [11], Artificial Bee Colony (ABC) [22], Harmony Search (HS) [23], and Fire Fly (FF) [24].

ISSN 2395-8618

B. Local Search

Local search algorithms are stochastic or deterministic methods that take as an input a solution generated randomly or by a specific algorithm, looking for transitions with the neighbors to this point at a given time. The goal is to find a better individual and to convert it in the next configuration, maintaining the original element if no improvement is detected [25]. The concept of vicinity is fundamental for LS since it represents the search area for individual refinement; in the case of combinatorial optimization this area is formed by the set of all solutions that can be reached by a unitary change in the current individual, while in continuous or numerical problems it is a dense set formed by an infinite number of points, and a modification strategy is required to find the neighbors [20].

Finding a good balance between the components of global and local search is one of the main design goals in a MA, and can be seen as an optimization process *per se*. From the way a memetic algorithm is compounded and in the implementation of its LS, it is seen that the equality MA = GS + LS is incorrect, since both searches are interrelated and are not designed as independent stages [14].

C. Classification of MAs

A MA can be classified as simple (canonical) or adaptive [26]. Simple MAs are characterized for *a priori* knowledge of the problem domain, that is incorporated to the algorithm design and produces a static behavior; in spite of belonging to the first generation of MAs, canonical hybrids are still popular for their easy implementation, particularly because of the use of genetic algorithms as global searchers.

On the other hand, adaptive MAs acquire information during their execution (learning), so they are able to reconfigure not only their parameters but their operators at run time, in order to adapt themselves to specific circumstances or instances of the problem [18]. The design of an adaptive MA requires to consider such aspects as the selection of subsets with agents for applying the fine search, the frequency and intensity of LS stages, the selection of procedures for the improvement, and the convergence of the population [6], [22].

III. MemMABC Algorithm

The Artificial Bee Colony is a swarm intelligence algorithm introduced by Karaboga as a method for numerical optimization [27], inspired on the behavior of bee hives in two natural processes: the recruitment of bees for the exploitation of food sources and the abandonment of exhausted sources. In ABC, the bees in a hive are divided in three groups: employed, onlookers and scouts, and each group represents an evaluation stage. There is an employed bee assigned to each source, and from this point the bee calculates a new solution and keeps the best of both. The number of onlookers is the same as the employed bees, and their assignation to sources is determined by the performance of such sources. The onlookers also calculate new solutions from their assigned source. Finally, when a source can't improve after a specified number of cycles, it is abandoned and replaced by a new one found by a scout bee.

Several versions of ABC have been developed; the modification proposed in [28], MABC, has an adaptation for constrained numerical optimization with a tournament-type selection based on the feasibility rules of Deb [29]. These criteria improve the process for selecting solutions at each iteration, by choosing the most feasible individual instead of the one with the best value for the objective function:

1) Between two feasible solutions, the best objective function value is preferred.
2) Between a feasible solution and another infeasible, the feasible is selected.
3) Between two infeasible solutions, the lowest sum of constraint violations (CVS) is preferred.

In this development a novel canonical MA, MemMABC, was designed taking as a base the MABC algorithm for global searching, with a modification to include a LS activated by time; Algorithm 1, *A1*, shows this memetic. The trigger for the LS stages is controlled by the variable $Frequency$, which indicates the period between an event of LS and the next one, in terms of a number of cycles or generations (line 39, A1). Although the iterations in original ABC and MABC are controlled by the number of cycles, MemMABC uses a variation to stop after a fixed number of objective function evaluations. This implementation permits a fair comparison [30] of MemMABC with other algorithms, specifically with MABC for the purposes of this work.

Algorithm 2, *A2*, shows the implementation of the local search method in MemMABC. It is a version of RW, modified to handle constraints by implementing the rules of Deb (line 21, A2). RW was chosen because of their easy implementation, since this method does not require the derivative of the objective function to calculate its gradient or its Hessian. Another modification was introduced to the original RW in order to reduce the computing effort and execution time, taking into account the complexity and high dimensionality of some real-world problems. RW requires the random generation of a number set R with values in the interval [-1,1], whose cardinality corresponds to the number of design variables, n. The values in R are transformed into search directions, so it is necessary to avoid a bias toward the diagonals of the unit hypercube surrounding the initial search point [31]. The random numbers generated are accepted only if $R < 1$, with R being computed as

$$R = (r_1^2 + r_2^2 + ... + r_n^2)^{1/2} \tag{1}$$

But when the problem to solve has a high dimensionality the algorithm can take several iterations to find a valid combination. This can be avoided if the elements on R are downsized when generated, using a constant divider (line 10, A2). The resultant array is a subset of R, so it is a valid combination. Finally, the LS depth or intensity is controlled by

ISSN 2395-8618

Algorithm 1. MemMABC

```
begin
    set Frequency, MaxEvs ;
    Evaluations = 0, g = 1;
    initialize the set of food sources x_i^0, i = 1, ..., SN;
    evaluate CVS and objective function for
    x_i^0, i = 1, ..., SN;
    if there are equality constraints then initialize ε(g);
    repeat
        if there are equality constraints then
            evaluate each x_i^0, i = 1, ..., SN with ε(g);
        end
        for I = 1 to SN do
            generate v_i^g with x_i^{g-1};
            evaluate CVS and objective function for v_i^g;
            Evaluations = Evaluations + 1 ;
            if v_i^g is better x_i^{g-1} (Deb's criteria) then
                x_i^g = v_i^g;
            else
                x_i^g=x_i^{g-1};
            end
        end
        for I = 1 to SN do
            select food source x_i^g based on binary
            tournament selection;
            generate v_i^g with x_i^g;
            evaluate CVS and objective function for v_i^g;
            Evaluations = Evaluations + 1 ;
            if v_i^g is better than x_i^{g-1} (Deb's criteria) then
                x_i^g = v_i^g;
            end
        end
        apply smart flight to those solutions whose limit
        to be improved has been reached;
        make Evaluations = Evaluations + 1 for
        every source improved ;
        keep the best solution so far x(Best)^g;
        g = g + 1;
        if there are equality constraints then update ε(g)
        if ((g)mod(Frequency)) == 0 then
            apply RW to x(Best)^g;
            TryLimit(Best)^g = 0;
        end
    until Evaluations >= MaxEvs;
end
```

Algorithm 2. Local search using RW

```
begin
    set MaxIter, MaxCount;
    take best solution so far as initial point X_0;
    evaluate CVS and objective function for X_0;
    Iterations = 1, Go = True, Evaluations = 1;
    while Go==True do
        repeat
            R = 0;
            for J = 1 to Var do
                Dir(J) = −0.1 + 2 ∗ rand(1, Var)/10;
                R = R + Dir(J)^2;
            end
            R = sqrt(R);
        until R < 1;
        for J = 1 to Var do
            U(J) = Dir(J)/R;
            X_Try(J) = X1(J) + λ ∗ U(J);
        end
        evaluate SVR and objective function for X_Try;
        Evaluations = Evaluations + 1 ;
        select best from X_Try and X_0 using Deb's
        criteria;
        if MaxIter < Iterations then
            λ = λ/2;
            Iterations = 1;
            if λ <= ε then
                Go = False;
            end
        else
            Iterations = Iterations + 1;
        end
        if Evaluations >= MaxCount then
            Go = False;
        end
    end
end
```

the parameter $MaxEvs$, who indicates the maximum number of evaluations for each activation.

IV. Analysis of the Four-bar Mechanism

A planar four-bar mechanism is formed by a reference bar (r_1), a crank or input bar (r_2), a coupler (r_3), and a rocker or output bar (r_4), as is shown in Figure 2. Two coordinate systems are proposed in order to analyze this mechanism: a system fixed to the real world (OXY) and another for self reference (OX_rY_r), where (x_0, y_0) is the distance between the origin of both systems, θ_0 is the rotation angle of the reference system and θ_i ($i = 2, 3, 4$) corresponds to the angles for the bars in the mechanism; finally, the coordinate pair (r_{cx}, r_{cy}) indicates the length of the support bars to position the coupler C.

A. Kinematics of the mechanism

The kinematics of four-bar mechanisms have been extensively treated, detailed explanations are in [32] and [33]. For analyzing the mechanism position the closed loop equation

ISSN 2395-8618

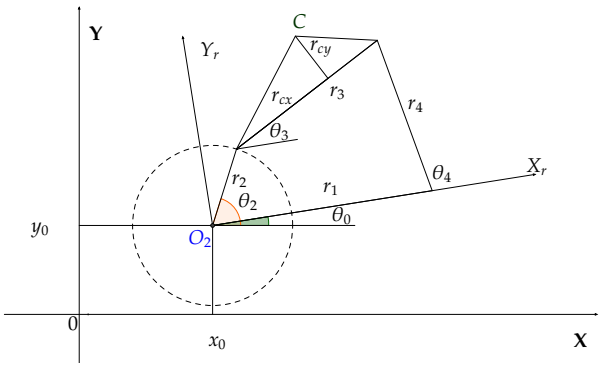


Fig. 2. Four-bar mechanism

can be established as:

$$\vec{r}_1 + \vec{r}_4 = \vec{r}_2 + \vec{r}_3 \tag{2}$$

Equation (2) can be expressed in polar notation as:

$$r_1 e^{j\theta_1} + r_4 e^{j\theta_4} = r_2 e^{j\theta_2} + r_3 e^{j\theta_3} \tag{3}$$

After applying Euler's equation to (3), the real and imaginary parts are:

$$\begin{aligned} r_1 cos\theta_1 + r_4 cos\theta_4 &= r_2 cos\theta_2 + r_3 cos\theta_3 \\ r_1 sin\theta_1 + r_4 sin\theta_4 &= r_2 sin\theta_2 + r_3 sin\theta_3 \end{aligned} \tag{4}$$

Left side of equation system (4) can be expressed in terms of θ_4 to obtain the angular position θ_3:

$$\begin{aligned} r_4 cos\theta_4 &= r_2 cos\theta_2 + r_3 cos\theta_3 - r_1 cos\theta_1 \\ r_4 sin\theta_4 &= r_2 sin\theta_2 + r_3 sin\theta_3 - r_1 sin\theta_1 \end{aligned} \tag{5}$$

The compact form of Freudenstein's equation is obtained by squaring the system (5) and adding its terms, as:

$$A_1 cos\theta_3 + B_1 sin\theta_3 + C_1 = 0 \tag{6}$$

with:

$$A_1 = 2r_3 \left(r_2 cos\theta_2 - r_1 cos\theta_1\right) \tag{7}$$

$$B_1 = 2r_3 \left(r_2 sin\theta_2 - r_1 sin\theta_1\right) \tag{8}$$

$$C_1 = r_1^2 + r_2^2 + r_3^2 - r_4^2 - 2r_1 r_2 cos\left(\theta_1 - \theta_2\right) \tag{9}$$

θ_3 can be obtained as a function of the parameters A_1, B_1, C_1 and θ_2, expressing $sin\theta_3$ and $cos\theta_3$ in terms of $tan\left(\theta_3/2\right)$ as follows:

$$sin\theta_3 = \frac{2tan(\theta_3/2)}{1 + tan^2(\theta_3/2)} \quad , \quad cos\theta_3 = \frac{1 - tan^2(\theta_3/2)}{1 + tan^2(\theta_3/2)} \tag{10}$$

A second-order lineal equation is obtained by substitution on (6):

$$\left[C_1 - A_1\right] tan^2\left(\frac{\theta_3}{2}\right) + \left[2B_1\right] tan\left(\frac{\theta_3}{2}\right) + A_1 + C_1 = 0 \tag{11}$$

Solving (11), the angular position θ_3 is given by:

$$\theta_3 = 2arctan\left[\frac{-B_1 \pm \sqrt{B_1^2 + A_1^2 - C_1^2}}{C_1 - A_1}\right] \tag{12}$$

B. Kinematics of the coupler

Since C is the point of interest in the coupler, to determine its position in the system OX_rY_r it has to be established that:

$$\begin{aligned} C_{xr} &= r_2 cos\theta_2 + r_{cx} cos\theta_3 - r_{cy} sin\theta_3 \\ C_{yr} &= r_2 sin\theta_2 + r_{cx} sin\theta_3 + r_{cy} cos\theta_3 \end{aligned} \tag{13}$$

Translated to the global coordinate system, this point is expressed as:

$$\begin{bmatrix} C_x \\ C_y \end{bmatrix} = \begin{bmatrix} cos\theta_0 & -sin\theta_0 \\ sin\theta_0 & cos\theta_0 \end{bmatrix} \begin{bmatrix} C_{xr} \\ C_{yr} \end{bmatrix} + \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \tag{14}$$

As can be observed, equations (13), (14) and the expressions corresponding to mechanism kinematics are sufficient to calculate the position of C along the trajectory.

V. Case Study

The problem addressed in this work is the length synthesis of a four-bar mechanism, designed for tracking a vertical linear trajectory indicated by a set of N precision points, without prescribed synchronization; that is, the point C of the coupler must *pass over* every point consecutively, without a pre-established sequence for such positions. The case study was selected because of its complexity; a measure for this complexity is the parameter ρ, which stands for the ratio between the feasible zone and the search space and can be represented by the percentage of feasible solutions found in an arbitrarily large set of randomly-generated vectors [34]. As the value of ρ diminishes, the computational effort required by the solving algorithm increases, since there are fewer available solutions. In order to evaluate ρ for this case a million of proposed solutions were taken, and only forty-three of them were feasible, resulting in a value of $\rho = 0.0043\%$.

Without loss of generality, the constrained optimization problem can be defined as to:

$$minimize \quad f(\vec{x}) \tag{15}$$

subject to:

$$g_j(\vec{x}) \leq 0, \qquad j = 1, 2, ..., p \tag{16}$$

$$h_k(\vec{x}) = 0, \qquad k = 1, 2, ..., q \tag{17}$$

where $\vec{x}$ is the vector of variables with dimension n, $f(\vec{x})$ is the objective function, $g_j(\vec{x})$ is the set of p inequality constraints and $h_k(\vec{x})$ is the set of q equality constraints.

A. Objective function

As a result of the mechanism synthesis will be calculated the length of the bars, the distance and rotation angle between the coordinate systems, and the angle set for the input bar. In the global coordinate system OXY, the precision point C_d^i is indicated as:

$$C_d^i = \left[C_{xd}^i, C_{yd}^i\right]^T \tag{18}$$

The set of precision points is defined as:

$$\Omega = \left\{C_d^i | i \in N\right\} \tag{19}$$

ISSN 2395-8618

Given a set of values of the mechanism bars and their parameters x_0, y_0, θ_0, each position of the coupler can be expressed as a function of the input bar angle:

$$C^i = \left[C_x(\theta_2^i), C_y(\theta_2^i)\right]^T \tag{20}$$

It is desired to minimize the distance between the calculated and the precision points, C^i and C_d^i respectively; the function in (21) is proposed to quantify this error:

$$f(\theta_2^i) = \sum_{i=1}^{N} \left[(C_{xd}^i - C_x^i)^2 + (C_{yd}^i - C_y^i)^2\right] \tag{21}$$

B. Design constraints

The fulfillment of the performance constraints related to dimensions and mobility criteria of the mechanism is fundamental in its design, since these limits establish the physical reproducibility and esthetic.

1) Sequence of input angles: The generation of a trajectory without prescribed synchronization requires an ascendant or descendant order of the crank angle values, in correspondence to each precision point. If the angle for the precision point i is denoted as θ_2^i, this order can be expressed as:

$$\theta_2^1 < \theta_2^2 < ... < \theta_2^N \tag{22}$$

where N is the number of precision points.

2) Grashof's law: It is a fundamental consideration of design, since it defines the criteria to ensure complete mobility for at least one link on a four-bar mechanism. This law establishes that *for a planar four-bar linkage, the sum of the shortest bar and the largest bar cannot be larger than the sum of the remaining bars, if a continual relative rotation between two elements is desired* [12]. If the lengths of the shortest and largest links are denoted as s and l respectively, with p and q indicating the remaining links, it is established that:

$$l + s \leq p + q \tag{23}$$

For this synthesis problem, Grashof's law is given by:

$$r_1 + r_2 \leq r_3 + r_4 \tag{24}$$

Consequently the following restrictions are required:

$$r_2 < r_3, \quad r_3 < r_4, \quad r_4 < r_1 \tag{25}$$

C. Design variables

Consider the vector of design variables for the four-bar mechanism, established as:

$$\vec{p} = [p_1, p_2, ..., p_{15}]^T \tag{26}$$

$$= \left[r_1, r_2, r_3, r_4, r_{cx}, r_{cy}, \theta_0, x_0, y_0, \theta_2^1, ..., \theta_2^6\right]^T \tag{27}$$

where the first four variables are the mechanism bar lengths, the following two values represent the length of the supporting bars of the coupler, the subsequent three variables indicate the relative position between the coordinate systems, and the last six values correspond to the angle sequence for the input bar.

D. Optimization problem

Consider the mono-objective numerical optimization problem described by (28) to (41), to obtain the dimensional synthesis of a four-bar mechanism for trajectory generation over *N* precision points:

$$min \quad f(\vec{p}) = \sum_{i=1}^{N} \left[(C_{xd}^i - C_x^i)^2 + (C_{yd}^i - C_y^i)^2\right] \tag{28}$$

subject to:

$$g_1(\vec{p}) = p_1 + p_2 - p_3 - p_4 \leq 0 \tag{29}$$

$$g_2(\vec{p}) = p_2 - p_3 \leq 0 \tag{30}$$

$$g_3(\vec{p}) = p_3 - p_4 \leq 0 \tag{31}$$

$$g_4(\vec{p}) = p_4 - p_1 \leq 0 \tag{32}$$

$$g_5(\vec{p}) = p_{10} - p_{11} \leq 0 \tag{33}$$

$$g_6(\vec{p}) = p_{11} - p_{12} \leq 0 \tag{34}$$

$$g_7(\vec{p}) = p_{12} - p_{13} \leq 0 \tag{35}$$

$$g_8(\vec{p}) = p_{13} - p_{14} \leq 0 \tag{36}$$

$$g_9(\vec{p}) = p_{14} - p_{15} \leq 0 \tag{37}$$

and the precision points:

$$\Omega = \{(20, 20), (20, 25), (20, 30), (20, 35), (20, 40), (20, 45)\} \tag{38}$$

with the bounds:

$$0 \leq p_i \leq 60, \quad i = [1, 2, 3, 4] \tag{39}$$

$$-60 \leq p_i \leq 60, \quad i = [5, 6, 8, 9] \tag{40}$$

$$0 \leq p_i \leq 2\pi, \quad i = [7, 10, 11, ..., 14, 15] \tag{41}$$

VI. Optimization Algorithm

Algorithms 1 and 2 correspond to the global and local search sections of the proposed memetic MemMABC, respectively. This algorithm requires six user-defined parameters: the number of sources or possible solutions SN, the maximum number of cycles or generations MCN, the maximum number of the objective function evaluations $MaxEvs$, the frequency of LS activation $Frequency$, the LS depth $MaxCount$, and the number of consecutive trials for improvement a source is kept before being replaced $TryLimit$, that is calculated as:

$$TryLimit = SN * n \tag{42}$$

where n is the number of design variables.

The implementation of the proposed algorithm was programmed in MATLAB R2013a, and the simulations were carried out on a computational platform with the following characteristics: Intel Core i7@2.6 GHZ microprocessor, 8Gb RAM memory and Windows 8 Operating system. All the algorithm simulations were executed with the following parameter values: $SN = 50$, $MCN = 8,000$, $MaxEvs = 1,000,000$, $Frequency = 1,750$, $MaxCount = 40,000$, and $TryLimit = 750$, calculated from (42).

ISSN 2395-8618

VII. Analysis of Results

Thirty independent runs were executed for the selected case study with both algorithms, MABC and MemMABC; a statistical analysis of their results is presented in Table I. As can be seen, the minimum value for the objective function was obtained with MemMABC ($OF = 0.014667757429261$), with a variance of $\sigma^2 = 0.086822826371684$, significantly lesser than the correspondent evaluation for MABC; since the variance measures the dispersion of a set of random variables with respect to their arithmetic mean, this value indicates a steady operation of the algorithm. Additionally, MemMABC required a 10% less evaluations to find its best result, even before reaching the stop condition given by $MaxEvs$. The results show the synergy formed by the combination of these global and local searchers, and the inclusion of a technique for handling of constraints.

TABLE I
Statistical Analysis of MABC vs MemABC

Parameter	MABC	MemMABC
Minimum	0.029598038968931	0.014667757429261
Maximum	6.047393204762160	1.324383428475970
Variance	1.158659308687650	0.086822826371684
Standard Dev	1.076410381168650	0.294657133583566
Evaluations Req	1,000,000	900,000

Figure 3 shows a simulation of the best-solution mechanism calculated by MemMABC, and its trajectory over the precision points, marked as $C_1, C_2, ..., C_6$. As it can be seen, the mechanism passes over the precision points in its ascending path and the return loop is quite small. Consequently the recovering time is short, a plus if the device is analyzed from an engineering point of view.

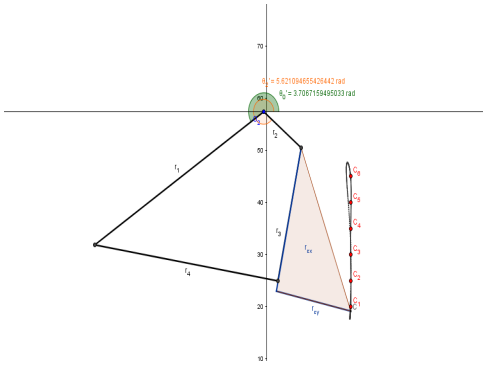


Fig. 3. Simulation of MemMABC best solution

A set with five solutions was selected considering the best values of the objective function; Table II present the solution vectors included in that set. As can be seen, all the values

TABLE II
Best five solution vectors

Variable	1	2	3	4	5
r_1	47.61022	47.65784	50.59788	50.10518	37.74315
r_2	11.29351	10.11871	12.73210	10.64155	11.63338
r_3	26.05979	25.87023	23.19406	26.15958	23.60134
r_4	44.17324	42.22803	47.15120	43.20891	37.09528
r_{cx}	28.09277	26.33869	21.27262	24.18547	28.33434
r_{cy}	18.09717	20.42422	15.87719	20.65887	7.90358
θ_0	3.70672	3.70295	3.57266	3.66212	3.80425
x_0	−0.72692	−1.06722	4.14226	0.71014	5.86156
y_0	57.39457	57.74773	53.35509	57.35837	58.09489
θ_2^1	2.03045	1.70228	2.09284	1.77039	1.02760
θ_2^2	2.54567	2.44986	2.55090	2.44951	2.14131
θ_2^3	2.97677	2.93643	2.94565	2.91994	2.58235
θ_2^4	3.40864	3.40469	3.35006	3.37768	3.02345
θ_2^5	3.87709	3.90614	3.79183	3.87046	3.53605
θ_2^6	4.45365	4.55469	4.30554	4.50137	4.21593
OF	0.01467	0.02635	0.02968	0.03634	0.04452

fell within the limits marked by design constraints; because of the table size, quantities are represented using only five decimal digits in spite of being calculated in the simulation with a precision of fourteen decimal places. The results generated after the simulations demonstrate the capability of MemMABC for balancing diversification over the area of feasible solutions and intensification for local exploitation.

VIII. Conclusion

A novel proposal of a memetic algorithm for solving real-world engineering design problems is presented in this paper, using a combination of algorithms with Modified Artificial Bee Colony and Random Walk. From the obtained results it is established that: 1) the algorithm improves the performance of MABC, not only in relation to the search for the optimal, but in reference to the stability of the search; 2) considering the point of view of engineering design, this optimization method produces good solutions since they are physically and esthetically reproducible; 3) no extensive computing resources are required for its implementation, and 4) it is a simple algorithm with an easy implementation. It should be mentioned that wide ranges of values for the design variables were used for simulating solutions of the proposed optimization problem, so results can be improved if such ranges are bounded more closely, accordingly to the physical specifications of the real model.

Although in this paper a specific case of synthesis for a four-bar mechanism is addressed, the simplicity of the proposed algorithm facilitates its use for the designing of other types of mechanisms and devices. In this sense, the main difficulty is an adequate interpretation and formulation of the particular problem and its corresponding constraints. The initial configuration of the algorithm requires special attention

ISSN 2395-8618

and is a line that can be the base for future work, considering both the previous tuning of parameters and their control during execution.

Finally, the main future work for this development is its transformation from a canonical memetic to an adaptive algorithm, with the capability to modify itself by incorporating knowledge to this process of self adaptation. This transformation implies the application of new techniques for local search, and to implement the intelligence required to process more than one meme for the learning.

Acknowledgment

The authors would like to thank Instituto Politécnico Nacional (IPN) for its support via Secretaría de Investigación y Posgrado (SIP) with the research project SIP-20151320.

References

[1] I. Boussaid, J. Lepagnot, and P. Siarry, "A survey on optimization metaheuristics," *Information Sciences*, vol. 237, pp. 82–117, 2013.

[2] X. Yang, *Harmony Search as a Metaheuristic Algorithm*. Springer Berlin, 2009, vol. 191, pp. 1–14.

[3] S. Domínguez Isidro, E. Mezura Montes, and G. Leguizamón, "Memetic differential evolution for constrained numerical optimization problems," in *Proceedings of 2013 IEEE Congress on Evolutionary Computation*, 2013, pp. 2996–3003.

[4] N. Krasnogor, A. Aragón, and J. Pacheco, *Memetic Algorithms*. Springer Berlin, 2012, vol. 2, ch. 12, pp. 905–936.

[5] E. Ozcan and C. Basaran, "A case study of memetic algorithms for constraint optimization," *Soft Computing*, vol. 13, pp. 871–882, 2009.

[6] X. Cai, Z. Hu, and Z. Fan, "A novel memetic algorithm based on invasive weed optimization and differential evolution for constrained optimization," *Soft Computing*, no. 17, pp. 1893–1910, 2013.

[7] Y. Li, B. Wu, L. Jiao, and R. Liu, "Memetic algorithm with double mutation for numerical optimization," in *IScIDE 2011*, 2012, pp. 66–73.

[8] P. Garg, "A comparison between memetic algorithm and genetic algorithm for the cryptoanalysis of simplified data encryption standard algorithm," *International Journal of Network Security & Its Applications (IJNSA)*, vol. 1, no. 1, pp. 33–42, April 2009.

[9] Y. Zhang, F. Gao, Y. Zhang, and W. A. Gruver, "Dimensional synthesis of a flexible gripper with a high degree of stability," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, vol. 2, October 1996, pp. 1025–1030.

[10] P. Merz and B. Freisleben, "Memetic algorithms for the traveling salesman problem," *Complex Systems*, no. 13, pp. 297–345, 2001.

[11] J. Ni, L. Li, F. Qiao, and Q. Wu, "A novel memetic algorithm and its application to data clustering," *Memetic Computation*, 2013.

[12] J. E. Shigley and J. J. Uicker, *Teoría de Máquinas y Mecanismos*. México: McGraw Hill, 1988.

[13] R. Norton, *Diseño de Maquinaria, una Introducción a la Síntesis y Análisis de Mecanismos y Máquinas*. México: McGraw Hill, 1995.

[14] P. Moscato and C. Cotta, "Una introducción a los algoritmos meméticos," *Revista Iberoamericana de Inteligencia Artificial*, vol. 19, pp. 131–148, 2003.

[15] P. A. Moscato, "On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms," Caltech, Technical Report 826, 1989.

[16] F. Neri and C. Cotta, *Handbook of Memetic Algorithms*. Springer Verlag, 2012, ch. A Primer on Memetic Algorithms.

[17] Y. Ong, M. Lim, and X. Chen, "Memetic computation: Past, present and future," *IEEE Computational Inteligence Magazine*, vol. 5, no. 2, 2010.

[18] X. Chen, Y. Ong, and K. C. Tan, "A multi-facet survey on memetic computation," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 5, pp. 591–607, 2011.

[19] K. Krasnogor and J. Smith, "A tutorial for competent memetic algorithms: Model, taxonomy, and design issues," *IEEE Transactions on Evolutionary Computation*, vol. 9, no. 5, pp. 474–488, October 2005.

[20] M. A. Montes de Oca, C. Cotta, and F. Neri, *Local Search*. Springer Verlag, 2012, vol. 379, pp. 29–42.

[21] J. Tang, M. Lim, and Y. Ong, "Diversity-adaptive parallel memetic algorithm for solving large scale combinatorial optimization problems," *Soft Computing*, vol. 11, no. 9, pp. 873–888, July 2007.

[22] S. D. P. Rakshit, A. Konar and A. Nagar, "ABC-TDQL: An adaptive memetic algorithm," in *Proceedings of 2013 IEEE Workshop on Hybrid Intelligent Models and Applications*, April 2013, pp. 35–42.

[23] ——, "A bee foraging-based memetic harmony search method," in *Proceedings of 2012 IEEE International Conference on Systems, Man and Cybernetics*, October 2012, pp. 184–189.

[24] I. J. Fister, X. Yang, I. Fister, and J. Brest, "Memetic firefly algorithm for combinatorial optimization," Research Gate (online), April 2012.

[25] P. Moscato and C. Cotta, *A Gentle Introduction to Memetic Algorithms*. Kluwer Academic Publishers, 2003, pp. 105–144.

[26] N. Z. Y.S. Ong, M.H. Lim and K. Wong, "Classification of adaptive memetic algorithms: A comparative study," *IEEE Transactions on Systems, Man and Cybernetics, Part B Cybernetics*, vol. 36, no. 1, pp. 141–152, February 2006.

[27] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Erciyes University, Technical Report TR06, 2005.

[28] E. Mezura Montes and O. Cetina Domínguez, "Empirical analysis of a modified artificial bee colony for constrained numerical optimization," *Applied Mathematics and Computation*, vol. 218, pp. 10 943–10 973, 2012.

[29] K. Deb, "An efficient constraint handling method for genetic algorithms," *Computer Methods in Applied Mechanics and Engineering*, vol. 186, pp. 311–338, 2000.

[30] M. Mernik, S. Liu, D. Karaboga, and M. Crepinsek, "On clarifying misconceptions when comparing variants of the artificial bee colony algorithm by offering a new implementation," *Information Sciences*, vol. 291, pp. 115–127, 2015.

[31] S. Rao, *Engineering Optimization: Theory and Practice*, 4th ed. John Wiley and Sons, 2009.

[32] R. Pérez, *Análisis de Mecanismos y Problemas Resueltos*. Alfaomega Grupo Editor S.A. de C.V., 2006.

[33] E. Vega-Alvarado, E. Santiago-Valentín, A. Sánchez-Márquez, A. Solano-Palma, E. A. Portilla-Flores, and L. Flores-Pulido, "Síntesis óptima de un mecanismo plano para seguimiento de trayectoria utilizando evolución diferencial," *Research in Computing Science*, vol. 72, pp. 85–89, 2014.

[34] J. Liang, T. Runarsson, E. Mezura Montes, M. Clerc, P. Suganthan, C. Coello, and K. Deb, "Problem definitions and evaluation criteria for the CEC 2006," Nanyang Technological University, Technical Report, 2006.

ISSN 2395-8618

An Efficient Iterated Greedy Algorithm for the Makespan Blocking Flow Shop Scheduling Problem

Nouha Nouri and Talel Ladhari

***Abstract*—We propose in this paper a Blocking Iterated Greedy algorithm (BIG) which makes an adjustment between two relevant destruction and construction stages to solve the blocking flow shop scheduling problem and minimize the maximum completion time (makespan). The greedy algorithm starts from an initial solution generated based on some well-known heuristic. Then, solutions are enhanced till some stopping condition and through the above mentioned stages. The effectiveness and efficiency of the proposed technique are deduced from all the experimental results obtained on both small randomly generated instances and on Taillard's benchmark in comparison with state-of-the-art methods.**

***Index Terms*—Blocking, flow shop, makespan, iterated greedy method.**

I. Problem Description

In the Blocking Flow Shop Scheduling Problem (BFSP), there is a finite set of N jobs that must be processed on M machines in the same order. Indeed, since there is no buffer storage between each consecutive pair of machines, intermediate queues of jobs waiting for their next process are not allowed. So, a job cannot leave its current machine till the next downstream machine is clear. This blocking state avoids progressing of other jobs on the blocked shop.

Furthermore, each job i $(i = 1, 2, ..., N)$ ready at time zero and requiring non-negative time p_{ij} as a processing delay has to be processed first on machine $M1$, then on machine $M2$ and so on till on machine Mm $(j = 1, 2, ..., M)$. That is the sequence in which the jobs are to be processed is identical for each machine. Besides, the processing of a given job at a machine cannot be interrupted once started. Each job can be processed only on one machine at a time and each machine can process at most one job at a time. Based on the above definitions, the final objective is to find out a sequence for processing all jobs on all machines so that its maximum completion time (makespan) is minimized. Formally, the BFSP aborted in this research is the $Fm|block|C_{max}$ in conformance with the classifications mentioned by Graham et al. [1]. The most popular eccentric work done on this problem is [2] who showed that the $F2|blocking|C_{max}$ instance may be reduced to a special case of the traveling salesman problem which may be solved in polynomial time using Gilmore and Gomory algorithm [3]. When the number of machines exceeds two $(m > 2)$, then the problem becomes strongly NP-hard [4]. The BFSP may be sketched in many real-life situations. We may cite the robotic cell [5], the iron and steel production [6], the manufacturing of concrete blocks and other.

As well, let $\Pi := (\pi_1, \pi_2, ..., \pi_N)$ be a possible solution for the BFSP, where π_i denotes the i^{th} job in the specific sequence; $d_{\pi_i,j}$ $(i = 1, 2, ..., N; j = 0, 1, 2, ..., M)$ defines the departure time of job π_i on machine j, where $d_{\pi_i,0}$ represents the time job π_i begins its processing on the first machine. The corresponding values of makespan of Π may then be calculated as $C_{max}(\Pi) = C_{\pi_N,M}(\Pi)$ in $O(nm)$, where $C_{\pi_i,M} = d_{\pi_i,M}$ is the completion time of job π_i on machine M that can be calculated generally using expressions presented in [7]. We choose in this work to refer to the method based on tails calculation to express the makespan of a given permutation as $C_{max}(\Pi) = f_{1,1}$ where $f_{i,j}$ defines the length of time between the latest loading time of operation o_{ij} and the end of the operations for $j : M, M-1, ..., 1$; and $f_{i,M+1}$ is the duration between the latest completion time of operation o_{iM} and the end of the operations [8]. Consequently, we obtain the following recursive equations:

$$f_{N,M+1} = 0$$

$$f_{N,j} = f_{N,j+1} + p_{Nj} \qquad j = M, M-1, ..., 2 \qquad (1)$$

$$f_{i,M+1} = f_{i+1,M} \qquad i = N-1, N-2..., 1 \qquad (2)$$

$$f_{i,j} = max\{f_{i,j+1} + p_{ij}, f_{i+1,j-1}\} \qquad i = N-1, ..., 1; \quad j = M, ..., 2 \qquad (3)$$

$$f_{i,1} = f_{i,2} + p_{i1} \qquad i = N, N-1, ..., 1 \qquad (4)$$

In the beyond recursion, the tails of the last job on every machine are calculated first, then the second last job, and so on up to the first job.

Due to the NP-hardiness of the BFSP, small number of methods have been proposed to solve it. They are

Manuscript received on February 23, 2016, accepted for publication on June 19, 2016, published on June 25, 2016.

Nouha Nouri is with the Ecole Supérieure des Sciences Economiques et Commerciales de Tunis, University of Tunis, Tunisia (e-mail: nouri.nouha@yahoo.fr).

Talel Ladhari is with the Ecole Supérieure des Sciences Economiques et Commerciales de Tunis, University of Tunis, Tunisia and College of Business, Umm Al-Qura University, Umm Al-Qura, Saudi Arabia (e-mail: talel_ladhari2004@yahoo.fr).

ISSN 2395-8618

ranged from exact methods to meta-heuristic ones. Some of the most important techniques are briefly presented in the following. Solving the BFSP using exact methods has attracted few attention in comparison with other original flow shop problems [9], [10], [11]. In [12], a lower bounding schemes were exposed, next an exact method based on the Branch-and-Bound (B&B) technique which uses a new compounded lower bounds was developed in [13]. Besides, a double B&B algorithm using the reversibility property of the problem is proposed in [14]. As well, two MBIP models and one B&B algorithm were lately presented to solve to optimality the BFSP under the total completion time measure [15]. Certainly, exact methods are unable to solve the problem within a reasonable computational time. Therefore, this incapacity explains the necessity to employ heuristics and meta-heuristics. As constructive heuristics, we may cite the Profile Fitting (PF) [16] and the Nawaz-Enscore-Ham (NEH) technique [17]. Some priority rules and tie breaking strategies were proposed in [18]. In [19] and [20] the NEH-WPT heuristic and a constructive and a GRASP-based heuristics for the BFSP were introduced respectively. Basically, the NEH-WPT sorts all jobs in a non-decreasing order of the sum of their processing times on all machines.

Afterward, meta-heuristics algorithms appear as a complement to their counterparts heuristics. The (Ron) algorithm was presented in [13] regarding the blocking constraints, and a Tabu Search (TS) and an enhanced TS techniques were used by Grabowski et al. In [21], we locate an Iterated Greedy (IG) method based on the insertion stage of the NEH. Under the total flow time criterion, we cite the hybrid modified global-best Harmony Search (hmgHS) algorithm and the Discrete Artificial Bee Colony algorithm (DABC_D) technique exposed in [19] and [22], respectively. Under C_{max} criterion, an effective Revised Artificial Immune Systems (RAIS) algorithm is proposed in [23], a three-phase algorithm presented in [24], and a Discrete Particle Swarm Optimization algorithm (DPSO) with self-adaptive diversity control was treated in [25]. Subsequently, we refer to the Memetic Algorithm (MA) in [26], the Iterated Local Search algorithm (ILS) coupled with a Variable Neighborhood Search (VNS) in [27], and the Blocking Genetic Algorithm (BGA) and Blocking Artificial Bee Colony (BABC) algorithms in [28]. Experimental results demonstrated that both of the two later proposed algorithms are more efficient in finding better solutions than all other leading techniques.

Now, among meta-heuristics, we focused on the IG algorithm which is being applied to many scheduling problems and subsequently to flow shop variants [29]. It is simple and effective: the approach applies constructive methods iteratively to a selected solution and then uses an acceptance criterion to decide whether the obtained solution substitutes the old one. Indeed, a sequence of solution is obtained using some destruction and construction stages. The destruction phase removes some elements from one selected solution. Next, in the construction phase, a new solution is created by reconstructing a complete solution using a greedy constructive heuristic, which reinserts the removed elements in some order to form a new complete sequence. Facultatively, a local search algorithm may be added to boost the constructed solution.

The basic steps of the IG algorithm are given as shown in Table I. After presenting the problem background, the rest of this paper is organized as follows: In Section 2, the Blocking IG algorithm (BIG) is stated. In Section 3, the computational results and comparisons are provided, and Conclusions are made together with future research direction in Section 4.

TABLE I
PSEUDO-CODE OF IG ALGORITHM

Begin
Generate initial solution Π_0;
Apply local search to Π_0, and *Put* the modified solution into Π_s; % Optional
Repeat
Π_d= Destruction(Π_s);
Π_c= Construction(Π_d);
Π_l= Local search(Π_c); % Optional
Π_f= Acceptance criterion(Π_s,Π_l);
Until termination condition met
End

II. SOLVING THE BLOCKING FLOW SHOP PROBLEM BASED ON THE ITERATED GREEDY ALGORITHM

In this section, the details of the BIG proposed for the problem under discuss is introduced. We recall that the key components of all existing heuristics for the BFSP have been developed based on the NEH heuristic which is made up of two stages: the first stage is the creation of the preliminary sequence of the jobs, and next comes the iterative process of insertion of the resulting jobs depending on the initial sequence obtained in phase 1.

A. Initial solution

To generate the initial solution we have used the PF-NEH(x) heuristic as in [28]. However, instead of generating x solutions at the end of the heuristic we choose only the permutation with the minimum objective value. As well, we employ the insertion-based local search to produce a neighboring solution. In this local search, a job is removed from its original position and reinserted in all other possible places. The local technique is applied with a probability P_{ls}. Next, if the objective value is enhanced then the solution is replaced. The final permutation Π^s thus obtained is the seed sequence.

B. Destruction and construction stages

On the basis of an initial solution, the destruction phase is applied. This stage begins with a complete solution Π^s and then extracts $[q * \Pi^s]$ randomly chosen jobs from Π^s in

ISSN 2395-8618

an iterative way. The degree of destruction q is in the range [0,1]. This creates two subsequences: the first one contains the removed jobs Π^r, and the second subsequence is the rest of the initial sequence obtained after removing some jobs Π^s.

Now, based on these resulting subsequences, in the construction phase a final solution Π^c is then reconstructed using a greedy constructive algorithm by reinserting the previously removed jobs in the order in which they were extracted.

The pseudo-codes of the destruction and construction steps are as in Table II and Table III.

TABLE II
PSEUDO-CODE OF DESTRUCTION STAGE (Π^s,q)

Begin
Stage 1: Set Π^r empty
Stage 2: Let $\Pi^q \leftarrow \Pi^s$
Stage 3: For $i = 1$ ***to*** $(q * |\Pi^q|)$ ***Do***
1) $\Pi^q \leftarrow$ remove a randomly selected job from Π^q
2) $\Pi^r \leftarrow$ include the removed job in Π^r
End

TABLE III
PSEUDO-CODE OF CONSTRUCTION STAGE (Π^q,Π^r)

Begin
Stage 1: Let $\Pi^c \leftarrow \Pi^q$
Stage 2: For $j = 1$ ***to*** $|\Pi^r|$ ***Do***
1) $\Pi^c \leftarrow$ best permutation obtained after inserting job π^r_j in all possible positions of Π^c
End

C. Acceptance criterion

Once a newly reconstructed solution has been obtained, an acceptance criterion is applied to decide wether it will replace the current incumbent solution or not. We consider the Simulated Annealing (SA) acceptance criteria that may be achieved by accepting worse solutions with a certain probability as used in [29], [30]. This acceptance criterion is used with a constant temperature value, which depends on the number of jobs, the number of machines, and on other adjustable parameter λ:

$$Tempt = \lambda * \frac{\sum_{i=1}^{N}\sum_{j=1}^{M} p_{ij}}{10 * M * N} \quad (5)$$

Let $Mksp(\Pi^s)$ and $Mksp(\Pi^c)$ be respectively the makespan values of the current incumbent solution and the new reconstructed solution. Also, let rand() be a function returning a random number sampled from a uniform distribution between 0 and 1.

If $Mksp(\Pi^c) \geq Mksp(\Pi^s)$ Then Π^c is accepted as the new incumbent solution if:

$$rand() \leq exp\{Mksp(\Pi^c) - Mksp(\Pi^s)/Tempt\} \quad (6)$$

D. Final BIG algorithm

Considering all previous subsections, the proposed BIG algorithm for the BFSP goes as in Table IV.

TABLE IV
PSEUDO-CODE OF BIG ALGORITHM

Begin
Stage 1: Set the parameters: P_{ls}, q, λ and MCN.
Stage 2: Obtain the initial solution using the PF-NEH(x) heuristic. Depending on the local probability rate P_{ls}, improve the initial solution using the insertion-based local search technique. Let the final permutation Π^s be the seed sequence.
Stage 3: Let $\Pi^* = \Pi^s$
Stage 4:
While termination condition is not met ***Do***
1) Π^q= Destruction-phase(Π^s, q)
2) Π^c= Construction-phase(Π^q)
3) $\Pi^{c'}$= Local-phase(Π^c, P_{ls})
4) If $Mksp(\Pi^{c'}) < Mksp(\Pi^s)$ Then
   a) $\Pi^s := \Pi^{c'}$
   b) If $Mksp(\Pi^s) < Mksp(\Pi^*)$ Then
      i) $\Pi^* := \Pi^s$
5) Else If $(rand() \leq exp\{Mksp(\Pi^s) - Mksp(\Pi^{c'})/Tempt\})$ Then
   a) $\Pi^s := \Pi^{c'}$
Stage 5: Return the best solution found Π^*
End

III. COMPUTATIONAL RESULTS

In the following, to confirm the effectiveness and competitiveness of BIG, its performances are compared against some leading methods in the literature. As usually done, we have used the Taillard instances [31] to test our technique. This benchmark include 120 problems of multiple sizes arranged into 12 subsets. Each subset entails ten instances with equal size (20*5, 20*10, 20*20, 50*5, 50*10, 50*20, 100*5, 100*10, 100*20, 200*10, 200*20, and 500*20) where the first number define the job size and the second one represent the machine size.

Each instance is independently run 10 times and in each run we compute the percentage relative difference (PRD) using the following expression:

$$RPD(A) = \frac{(Mksp^A - Mksp^{Min}) \times 100}{Mksp^{Min}} \quad (7)$$

ISSN 2395-8618

where, $Mksp^A$ defines the value of the makespan reached by the BIG algorithm; and $Mksp^{Min}$ defines the minimum mekespan value obtained among all the compared algorithms.

The BIG algorithm is coded in $C++8.0$ and the experiments are executed on an Intel Pentium IV 2.4 GHz PC with 512 MB of memory.

The final experimental setup is given in Table V where the main purpose of the experiment was to compare the optimization performances of the algorithm under various system conditions.

TABLE V
THE EXPERIMENTAL SETUP

Factors	P_{ls}	q	λ	MCN
Values	0.2	0.3	2	100

A. Results on randomly generated instances

Before testing BIG algorithm on benchmark sets, the computational experiments have been at first carried out on a set of randomly generated instances obtained following the procedure explained in Taillard.

In our tests, the problem sizes are determined by varying the number of jobs and machines from 10 jobs and 3 machines to 100 jobs and 10 machines as was the case in [28].

This choice is fixed such to make comparison between BIG and BGA algorithm under this type of instances. Next, the C_{max} values of the best-found solutions for these generated instances were memorized for each of the compared heuristics.

A statistic for the solution quality for each set is given (Average RPD (ARPD)) as in Table VI. According to the above table, the proposed algorithm is more likely to get better solutions than BGA which is outperformed. For small instances, the two algorithms behave in the same way. Difference is observed by increasing the number of jobs.

TABLE VI
ARPD ON RANDOMLY GENERATED INSTANCES

Inst	BIG	BGA
10×3	0,000%	0,000%
10×5	0,000%	0,000%
10×7	0,000%	0,000%
20×3	0,000%	0,000%
20×5	0,000%	0,000%
20×7	0,000%	0,000%
50×3	0,000%	0,000%
50×5	0,007%	0,026%
50×7	0,000%	0,013%
70×3	0,010%	0,005%
70×5	0,005%	0,063%
70×7	0,000%	0,103%
100×3	0,007%	0,026%
100×5	0,006%	0,043%
100×7	0,000%	0,077%
Avrg	0,002%	0,024%

B. Comparing BIG with leading heuristics

In this subsection, we enlarge the domain of comparison and consider the BIG versus IG [21], MA [26], RAIS [23], and BGA [28] algorithms.

From Table VII, we can observe that the proposed BIG gives the best performance in terms of the overall solution quality, since it yields the minimum overall mean ARPD value equal to $0,041\%$, which is much better than those by the IG (0.744%), MA (0.174%), RAIS (0.426%), and BGA (0.055%).

More specifically, the BIG gives much better APRD than all compared heuristics and improves 86 out of 120 best-known solutions of Taillard's instances for the BFSP with the makespan criterion. The worst results are given by the IG [21].

Indeed, BIG algorithm behaves much more effective than the BIG algorithm as the size of instances increases. So, regardless its simplicity, we may assert that the BIG algorithm is an efficient heuristic in solving the BFSP and so may be used as a basis of comparison for future research.

TABLE VII
ARPD ON TAILLARD INSTANCES

Inst	BIG	MA	IG	RAIS	BGA
20×5	0,000%	0,000%	0,000%	0,000%	0,000%
20×10	0,000%	0,000%	0,000%	0,000%	0,000%
20×20	0,000%	0,000%	0,000%	0,000%	0,000%
50×5	0,022%	0,238%	0,322%	0,129%	0,032%
50×10	0,003%	0,199%	0,402%	0,203%	0,025%
50×20	0,005%	0,046%	0,267%	0,263%	0,030%
100×5	0,032%	0,572%	0,936%	0,109%	0,050%
100×10	0,027%	0,325%	1,032%	0,141%	0,058%
100×20	0,004%	0,245%	0,962%	0,242%	0,032%
200×10	0,000%	0,062%	0,631%	0,299%	0,015%
200×20	0,394%	0,052%	1,576%	0,936%	0,411%
500×20	0,001%	0,349%	2,805%	2,789%	0,011%
Avrg	0,041%	0,174%	0,744%	0,426%	0,055%

IV. CONCLUSION AND FUTURE WORK

In our study, BIG algorithm is proposed to solve the BFSP under makespan measure. This greedy method is very simple, and hybridized with a form of local search, enhanced much more the solutions quality.

The algorithm is developed to solve both randomly generated instances and a number of test problems (Taillard instances). The experiment results attest that BIG is better than other leading algorithms on all group instances specifically on high dimensional problems.

In the future, we will hybridize our technique using some hybrid evolutionary heuristics such as SA to improve its performance and design some better NEH heuristic variant to improve its efficiency.

ISSN 2395-8618

REFERENCES

[1] R. Graham, E. Lawler, J. Lenstra, and K. Rinnooy, "Optimization and approximation in deterministic sequencing and scheduling: A survey," *Annals of Discrete Mathematics*, vol. 5, pp. 287–362, 1979.

[2] S. Reddi and C. Ramamoorthy, "On the flow-shop sequencing problem with no wait in process," *Operational Research Quarterly*, vol. 3, pp. 323–31, 1972.

[3] P. Gilmore and R. Gomory, "Sequencing a one state variable machine: A solvable case of the traveling salesman problem," *Operations Research*, vol. 5, pp. 655–79, 1964.

[4] N. Hall and C. Sriskandarajah, "A survey of machine scheduling problems with blocking and no-wait in process," *Operations Research*, vol. 44, pp. 510–25, 1996.

[5] S. Sethi, C. Sriskandarajah, G. Sorger, J. Blazewicz, and W. Kubiak, "Sequencing of parts and robot moves in a robotic cell," *International Journal of Flexible Manufacturing Systems*, vol. 4, pp. 331–358, 1992.

[6] H. Gong, L. Tang, and C. Duin, "A two-stage flowshop scheduling problem on batching machine and a discrete machine with blocking and shared setup times," *Computers and Operations Research*, vol. 37, pp. 960–4, 2010.

[7] M. Pinedo, *Scheduling: theory, algorithms, and systems*. USA: Pretice Hall, 2008.

[8] L. Wang, Q. Pan, P. Suganthan, W. Wang, and Y. Wang, "A novel hybrid discrete differential evolution algorithm for blocking flowshop scheduling problems," *Computers and Operational Research*, vol. 3, pp. 509–20, 2010.

[9] E. Levner, "Optimal planning of parts machining on a number of machines," *Automation and Remote Control*, vol. 12, pp. 1972–8, 1969.

[10] I. Suhami and R. Mah, "An implicit enumeration scheme for the flowshop problem with no intermediate storage," *Computers and Chemical Engineering*, vol. 2, pp. 83–91, 1981.

[11] S. Karabati and P. Kouvelis, "Cycle scheduling in flow lines: modeling observations, effective heuristics and a cycle time minimization procedure," *Naval Research Logistics*, vol. 2, pp. 211–31, 1996.

[12] D. Ronconi and V. Armentano, "Lower bounding schemes for flowshops with blocking in-process," *Journal of the Operational Research Society*, vol. 11, pp. 1289–97, 2001.

[13] D. Ronconi, "A branch-and-bound algorithm to minimize the makespan in a flowshop problem with blocking," *Annals of Operations Research*, vol. 1, pp. 53–65, 2005.

[14] R. Companys and M. Mateo, "Different behaviour of a double branch-and-bound algorithm on $fm|prmu|cmax$ and $fm|block|cmax$ problems," *Computers and Operations Research*, vol. 34, pp. 938–953, 2007.

[15] G. Moslehi and D. Khorasanian, "Optimizing blocking flowshop scheduling problem with total completion time criterion," *Computers and Operations Research*, vol. 40, pp. 1874–1883, 2013.

[16] S. McCormick, M. Pinedo, S. Shenker, and B. Wolf, "Sequencing in an assembly line with blocking to minimize cycle time," *Operations Research*, vol. 37, pp. 925–935, 1989.

[17] M. Nawaz, J. Enscore, and I. Ham, "A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem," *Omega*, vol. 11, pp. 91–95, 1983.

[18] R. Companys, I. Ribas, and M. Mateo, "Note on the behaviour of an improvement heuristic on permutation and blocking flow-shop scheduling," *International Journal of Manufacturing Technology and Management*, vol. 20, pp. 331–57, 2010.

[19] L. Wang, Q. Pan, and M. Tasgetiren, "Minimizing the total flow time in a flowshop with blocking by using hybrid harmony search algorithms," *Expert Syst. Appl*, vol. 12, pp. 7929–7936, 2010.

[20] D. Ronconi and L. Henriques, "Some heuristic algorithms for total tardiness minimization in a flowshop with blocking," *Omega*, vol. 2, pp. 272–81, 2009.

[21] I. Ribas, R. Companys, and X. Tort-Martorell, "An iterated greedy algorithm for the flowhsop scheduling problem with blocking," *Omega*, vol. 3, pp. 293–301, 2011.

[22] G. Deng, Z. Xu, and X. Gu, "A discrete artificial bee colony algorithm for minimizing the total flow time in the blocking flow shop scheduling," *Chinese Journal of Chemical Engineering*, vol. 20, pp. 1067–1073, 2012.

[23] S. Lin and K. Ying, "Minimizing makespan in a blocking flowshop using a revised artificial immune system algorithm," *Omega*, vol. 41, pp. 383–389, 2013.

[24] Q. Pan and L. Wang, "Effective heuristics for the blocking flowshop scheduling problem with makespan minimization," *Omega*, vol. 2, pp. 218–29, 2012.

[25] X. Wang and L. Tang, "A discrete particle swarm optimization algorithm with self-adaptive diversity control for the permutation flowshop problem with blocking," *Applied Soft Computing*, vol. 12, pp. 652–662, 2012.

[26] Q. Pan, L. Wang, H. Sang, J. Li, and M. Liu, "A high performing memetic algorithm for the flowshop scheduling problem with blocking," *IEEE Transactions on Automation Science and Engineering*, vol. 10, pp. 741–756, 2013.

[27] I. Ribas, R. Companys, and X. Tort-Martorell, "An efficient iterated local search algorithm for the total tardiness blocking flow shop problem," *International Journal of Production Research*, vol. 51, pp. 5238–5252, 2013.

[28] N. Nouri and T. Ladhari, "Minimizing regular objectives for blocking permutation flow shop scheduling: Heuristic approaches," in *GECCO 2015*, 2015, pp. 441–448.

[29] R. Ruiz and T. Stützle, "A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem," *European Journal of Operational Research*, vol. 177, pp. 2033–49, 2007.

[30] ——, "An iterated greedy heuristic for the sequence dependent setup times flowshop problem with makespan and weighted tardiness objectives," *European Journal of Operational Research*, vol. 187, pp. 1143–1159, 2008.

[31] E. Taillard, "Benchmarks for basic scheduling problems," *European Journal of Operational Research*, vol. 64, pp. 278–85, 1993.

ISSN 2395-8618

Journal Information and Instructions for Authors

I. Journal Information

Polibits is a half-yearly open-access research journal published since 1989 by the *Centro de Innovación y Desarrollo Tecnológico en Cómputo* (CIDETEC: Center of Innovation and Technological Development in Computing) of the *Instituto Politécnico Nacional* (IPN: National Polytechnic Institute), Mexico City, Mexico.

The journal has double-blind review procedure. It publishes papers in English and Spanish (with abstract in English). Publication has no cost for the authors.

A. Main Topics of Interest

The journal publishes research papers in all areas of computer science and computer engineering, with emphasis on applied research. The main topics of interest include, but are not limited to, the following:

- Artificial Intelligence
- Natural Language Processing
- Fuzzy Logic
- Computer Vision
- Multiagent Systems
- Bioinformatics
- Neural Networks
- Evolutionary Algorithms
- Knowledge Representation
- Expert Systems
- Intelligent Interfaces
- Multimedia and Virtual Reality
- Machine Learning
- Pattern Recognition
- Intelligent Tutoring Systems
- Semantic Web
- Robotics
- Geo-processing
- Database Systems
- Data Mining
- Software Engineering
- Web Design
- Compilers
- Formal Languages
- Operating Systems
- Distributed Systems
- Parallelism
- Real Time Systems
- Algorithm Theory
- Scientific Computing
- High-Performance Computing
- Networks and Connectivity
- Cryptography
- Informatics Security
- Digital Systems Design
- Digital Signal Processing
- Control Systems
- Virtual Instrumentation
- Computer Architectures

B. Indexing

The journal is listed in the list of excellence of the CONACYT (Mexican Ministry of Science) and indexed in the following international indices: Web of Science (via SciELO citation index), LatIndex, SciELO, Redalyc, Periódica, e-revistas, and Cabell's Directories.

There are currently only two Mexican computer science journals recognized by the CONACYT in its list of excellence, *Polibits* being one of them.

II. Instructions for Authors

A. Submission

Papers ready for peer review are received through the Web submission system on www.easychair.org/conferences/?conf=polibits1; see also updated information on the web page of the journal, www.cidetec.ipn.mx/polibits.

The papers can be written in English or Spanish. In case of Spanish, author names, abstract, and keywords must be provided in both Spanish and English; in recent issues of the journal you can find examples of how they are formatted.

The papers should be structures in a way traditional for scientific paper. Only full papers are reviewed; abstracts are not considered as submissions. The review procedure is double-blind. Therefore, papers should be submitted without names and affiliations of the authors and without any other data that reveal the authors' identity.

For review, a PDF file is to be submitted. In case of acceptance, the authors will need to upload the source code of the paper, either Microsoft Word or LaTeX with all supplementary files necessary for compilation. Upon acceptance notification, the authors receive further instructions on uploading the camera-ready source files.

Papers can be submitted at any moment; if accepted, the paper will be scheduled for inclusion in one of forthcoming issues, according to availability and the size of backlog.

See more detailed information at the website of the journal.

B. Format

The papers should be submitted in the format of the IEEE Transactions 8x11 2-column format, see http://www.ieee.org/publications_standards/publications/authors/author_templates.html. (while the journal uses this format for submissions, it is in no way affiliated with, or endorsed by, IEEE). The actual publication format differs from the one mentioned above; the papers will be adjusted by the editorial team.

There is no specific page limit: we welcome both short and long papers, provided that the quality and novelty of the paper adequately justifies its length. Usually the papers are between 10 and 20 pages; much shorter papers often do not offer sufficient detail to justify publication.

The editors keep the right to copyedit or modify the format and style of the final version of the paper if necessary.

See more detailed information at the website of the journal.