"Vasile Alecsandri" University of Bacău
Faculty of Sciences
Scientific Studies and Research
Series Mathematics and Informatics
Vol. 21 (2011), No. 1, 185-192

# BIO-INSPIRED COMPONENTS FOR A BANDWIDTH PROBLEM

CAMELIA-M. PINTEA and ANDREEA VEŞCAN

**Abstract.** In this paper are illustrating some specific bio-inspired component for solving a combinatorial optimization problem: The *Matrix Bandwidth Minimization Problem(MBMP).* The described components are based on a hybrid model of the *Ant Colony System* technique with new local search mechanisms [4]. *MBMP* seeks for a simultaneous permutation of the rows and also of the columns of a square matrix in order to keep its nonzero entries close to the main diagonal.

## 1. INTRODUCTION

*Component Based Development (CBD)* is an active area of research and development. *CBSE* covers both component development and system development with components [1]. Development using components is focused on the identification of reusable entities and relations between them, starting from the system requirements.

Complexity of nowadays problems leads to a large number of approaches in order to solve these difficult problems. One of these problems is the *Matrix Bandwidth Minimization Problem (MBMP)*. It is a *NP*-complete [14] problem. The applications of the *MBMP* are in various fields as economics, physics and computer science and in engineering field. Other approaches are already developed with *GRASP* [8], *Tabu Search* [7], *GA* [9], a node-shift heuristic [7]. Based on *MBMP*, new problems were defined (see the antibandwidth problem [6]).

**Keywords and phrases:** Simulation and modeling, Model Development, Bandwidth Problem, Ant Colony.
**(2010) Mathematics Subject Classification:** 68N30, 68T20.

It is proposed a new approach to address the matrix bandwidth minimization problem using components. The paper is organized as follows: the investigated matrix bandwidth minimization problem is briefly presented in Section 2; the algorithms are described in Section 3; the proposed component-based solution is discussed in Section 4, and the last section describes the future development of our work.

## 2. THE BANDWIDTH PROBLEM

The *Matrix Bandwidth Minimization Problem (MBMP)* follows.

Given a square symmetric matrix *A* of order n, its bandwidth is $\beta = \max_{a_{ij} \neq 0} |i - j|$.

Solving the *MBMP* for *A* means finding a permutation $\pi$ of the rows and columns of the matrix *A*, that minimizes the bandwidth of the resulted matrix. *MBMP* as a graph problem follows.

Starting from *A* one can define the graph $G_A = (V, E)$ with $V = \{1, 2, ..., n\}$ and $E = \{(i, j) \ \textit{iff} \ a_{ij} \neq 0\}$. This leads to the bandwidth $\beta = \max_{(i,j)\in E} |i - j|$ of $G_A$. Solving the transformed problem means to find a permutation $\pi$ of *V* minimizing the graph bandwidth.

## 3. HYBRID MODEL FOR SOLVING MBMP

In the ant-based model for solving *MBMP, Ant Colony System* is hybridized with a local search mechanism [4]. The main steps of the algorithm, based on the level structure described by the *Cuthill-McKee* algorithm are following:

- computing the current matrix bandwidth and setting the parameters values;
- the construction phase follows, starting with all the ants in the node from the first level and successively making pseudo-randomly choices from the available neighbors.

  The local update rule is applied [5] after each step. This phase ends with the global pheromone update rule [5].
- writing the best solution.

These phases are iteratively executed within a given number of iterations. A solution is a one-dimensional array with the permutation of *V = {1, 2, · · · , n}*. Furthermore, the integration of a local search phase within the proposed *ACS* approach to *MBMP* facilitates the refinement of ants solutions. The pseudocode of the proposed model, as in [4] is given below:

**Hybrid ACS model for solving MBMP**
**begin**
**I. Initialization**: computes the current matrix bandwidth;
initialize pheromone trails; sets the parameters values;
**II. while** (maximum number of iterations not reached) **do**
**swap procedure**
**while** (maximum number of ants not reached) **do**
build a partial solution using ACS
apply a local pheromone update rule
**swap procedure**
apply a global pheromone update rule
**end while**
**end while**
**III. write** the best solution
**end**

For the improvement of *ACS* solutions with a specific local search mechanism, it is introduced the *PSwap* procedure (see paper [4] for details) with the aim of reducing the maximal bandwidth. This procedure is used twice within the proposed hybrid model: at the beginning of the iteration and after each partial solution is built, in order to improve each ant's solution. The hybrid *ACS* model based on *Swap Procedure: PSwap* as the local search stage is denoted *hACS*.

## 4. COMPONENT-BASED ANT COLONY SYSTEM FOR MBMP

This section presents the architecture of the component-based approach for the *MBMP,* the control flow and the data flow model. At the end of this section we have illustrated how the computation steps are successively executed.

We must first establish our entities involved in the component system definition before describing our component-based approach for modeling *MBMP*.

We can view components from a different perspective [2] as simple components and compound components. The wiring of components [3] in order to construct a component-based system is made using a connection between the output of a component and the input of another component. The execution of the *BlackBox* component [3] is composed of sequences of the form (operation, component), where operation is propagation (→) or evaluation (≡).

### 4.1 MBMP architecture

In section 3 the hybrid ant-based system for the *Matrix Bandwidth Minimization Problem* is described. Based on the pseudocode we can illustrate the algorithm using components (Figure 1) as a first level of design: reading input data, computation and printing the results.

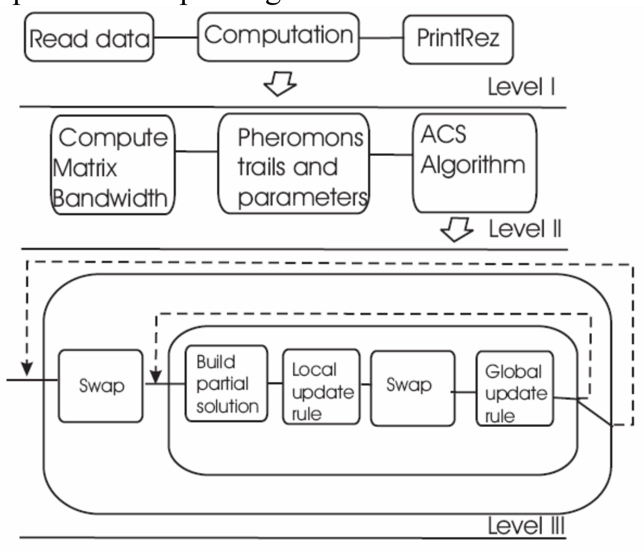


**Figure 1. Architectural levels**

The second design level includes: the computational part of the current matrix bandwidth, the pheromone trails initialization. Here are also set the parameters and the hybrid ant algorithm.

The third design level illustrates in detail the hybrid ant colony system algorithm: building partial solutions, applying the local pheromone update rule for all ants, applying a global pheromone update rule and the swap procedures described in Section 3.

All these computations are performed for tmax steps/times.

### 4.2 General and internal computation steps

The steps of the computation of the hybrid ant colony component model for *MBMP* are described successively.

We denote each component from the architecture (see figure 1 for details) with the following acronyms in order to be easier to read: *Readdata* component with *RD*; *ComputeMatrixBandwidth* with *CMB*;

*Pheromonstrailsandparameters* with *PTP*; *Swap* with *SW*; *Buildpartialsolution* with *BPS*; *Localupdaterule* with *LUR; Globalupdaterule* with *GUR* and *Printrez* with *PR*.

General computational steps for the representation 1:

• *state0 = ({RD ≡}, {RD});*
• *state1 = ({RD →}, {});*
-data is propagated throw the connections to the *PTP* component;
• *state2 = ({PTP ≡}, {PTP});*
-data is generated; initializing the pheromone trails;
-setting the parameters values.
• *state3 = ({PTP →}, {});*
-*PTP →*: data is propagated throw the connections to the *SW* component.
• *state4 = ({SW ≡}, {SW});*
-SW ≡: the swap procedure.
• *state5 = ({SW →}, {});*
-*SW →* data is propagated throw the connection to the input of the *BPS* component.
• *state6 = ({BPS ≡}, {BPS});*
-*BPS ≡:* is building the partial solution.
• *state7 = ({BPS →}, {});*
-*BPS →* data is propagated throw the connection to input of the *LUR* component.
• *state8 = ({LUR ≡}, {LUR});*
-*LUR ≡:* is applied the local update rule;
• *state9 = ({LUR →}, {});*
-*LUR →* data is propagated throw the connection to input of the *SW* component.
• *state10 = ({SW ≡}, {SW});*
-*SW ≡:* the swap procedure;
• *state11 = ({SW →}, {});*
-*SW →* data is propagated throw the connection to input of the *GUR* component.
• *state12 = ({GUR ≡}, {GUR});*
-*GUR ≡:* is applied the global update rule;
• *state13 = ({GUR →}, {});*
-*GUR →* data is propagated throw the connection to the *BPS* component or to the *SW* component or to the *PR* component, the decision is based on the number of iterations or to the number of ants as described in Section 3. If the *BPS* component is selected, then the computation starts from state6, if the

component *SW* is selected the computations start from state4. If the *PR* component is selected then the next state will be state14.

• *state14 = ({PR ≡}, {PR});*

-*PR ≡:* prints the results.

• *state15 = ({}, {}).*

-There are no more possibilities of applying either propagation or evaluation.

-The execution of the components involved in the system is finished.

## 5. CONCLUSIONS

Bio-inspired algorithms and in particularly ant-based algorithms are nowadays certified heuristics for solving combinatorial optimization problems. Components based on a hybrid *Ant Colony System* for the Matrix Bandwidth Minimization Problem are described. The way of using the components and the execution steps are illustrated.

## REFERENCES

1. I. Crnkovic, **Component-based Software Engineering - New Challenges in Software Development**, Software Focus, 2001.
2. A. Fanea, S. Motogna, **A Formal Model for Component Composition, Proceedings of the Symposium Zilele Academice Clujene**, 160–167, 2004.
3. A. Fanea, **Specification, Construction and Execution of a Component-Based Model,** Proceeding of the Symposium Colocviul Academic Clujean de Informatica, 87–92, 2005.
4. C-M. Pintea, G-C. Crisan, C.Chira, **A Hybrid ACO Approach to the Matrix Bandwidth Minimization Problem**, Hybrid Artificial Intelligent Systems, Part I, LNCS/LNAI 6076, 405-412 Springer, 2010
5. Dorigo, M., Stüzle, T.: **Ant Colony Optimization**. MIT Press, Cambridge (2004)
6. Leung, J.Y.-T., Vornberger, O., Witthoff, J.D.: **On some variants of the bandwidth minimization problem**. SIAM J. on Computing 13, 650-667 (1984)
7. Lim, A., Rodrigues, B., Xiao, F.: **Heuristics for matrix bandwidth reduction.** European J. of Oper. Res. 174, 1, 69-91 (2006)
8. Piana, E., Plana, I., Campos, V., Mart, R.: **GRASP and Path Relinking for the matrix bandwidth minimization**, European J. of Oper. Res. 153, 1, 200-210 (2004)

9. Rodriguez-Tello, E., Jin-Kao, H., Torres-Jimenez, J.: **An improved Simulated Annealing Algorithm for the matrix bandwidth minimization**. European J. of Oper. Res. 185,3, 1319-1335 (2008)

Camelia-M. Pintea,
"G. Coşbuc" National College, Cluj-Napoca, Romania,
cmpintea@yahoo.com

Andreea Vescan,
Department of Computer Science
Faculty of Mathematics and Computer,
Babes-Bolyai University, Cluj-Napoca, Romania
avescan @cs.ubbcluj.ro