The Band Collocation Problem: a Library of Problems and a Metaheuristic Approach

Hakan Kutucu[1], Arif Gursoy[2], Mehmet Kurt[3], and Urfat Nuriyev[2]

[1] Karabuk University, Department of Computer Engineering, Karabuk, Turkey
hakankutucu@karabuk.edu.tr,
[2] Ege University, Department of Mathematics, Izmir, Turkey
arif.gursoy@ege.edu.tr, urfat.nuriyev@ege.edu.tr
[3] Izmir University, Department of Mathematics and Computer Science, Izmir, Turkey
mehmet.kurt@izmir.edu.tr

Abstract. In this paper, we consider the Band Collocation Problem (BCP) which may find an application in telecommunication networks, to design an optimal packing of information flows on different wavelengths into groups for obtaining the highest available cost reduction using wavelength division multiplexing (WDM) technology. We give a review of its mathematical models. The linear and nonlinear models have been implemented in GAMS (the General Algebraic Modeling System) and solved using the CPLEX and KNITRO solvers, respectively. Then, we introduce the BCP Library (BCPLib) including 1296 problem instances with different properties that can be accessed at http://www.izmir.edu.tr/bps. Finally, we improve a simulated annealing (SA) meta-heuristic to solve the BCP. The proposed algorithm is performed using two local search methods for several test instances of the BCPLib and compared with the solutions obtained by a genetic algorithm. Experimental results showed that the proposed algorithm improves the quality of solutions.

Keywords: Bandpass problem, Combinatorial optimization, Mathematical modeling, Telecommunication, Simulated annealing

1 Introduction

The Bandpass Problem (BP) whose first mathematical model was presented in 2009 is a combinatorial optimization problem which may be used in telecommunication systems [1]. Due to the development in the technology, some problems become invalid or useless and they need to be updated. In this sense, Nuriyev et. al. announced the Band Collocation Problem (BCP) by extending the BP due to incompatibility with real life implementations at the present time [7].

The BP is related to transmitting data over fiber optic networks using the Dense Wavelength Division Multiplexing (DWDM) technology [1]. The data is transmitted from a source to other stations on different wavelengths in a single fiber optic cable.

In: A. Kononov et al. (eds.): DOOR 2016, Vladivostok, Russia, published at http://ceur-ws.org

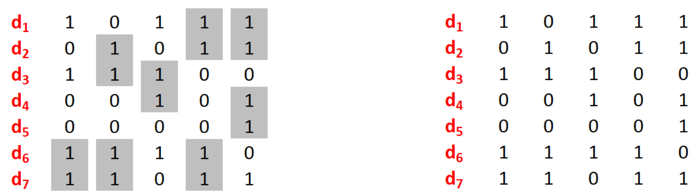
Stations add/drop data onto/from the cable via an optical device called Add/Drop Multiplexers (ADM). Special cards in ADMs control each wavelength. They can add/drop (extract) data at some wavelengths to/from a network path [5]. Stations do not have to receive all data on the cable.

Each special card is normally responsible for one wavelength. However, according to the BP, a practical programmable ADM can add/drop multiple wavelengths if they are neighboring to each other, that is, if they are consecutive. In the BP, a group of consecutive wavelengths is called bandpass and the length of a bandpass is represented by a positive integer B called bandpass number.

Companies want to reduce the costs of constructing the network. Actually, this is the goal of the BP by maximizing the number of bandpasses. The key idea of the BP is to gather up requested data's wavelengths at any station.

In order to model this problem, we use a binary matrix corresponding to the network traffic. Consider a communication network. The communication is conducted on m different wavelengths to carry data to be sent to n different stations. This situation is described by a binary matrix $A = a_{ij}$: if data carried on wavelength $i = 1, ..., m$ is requested by station $j = 1, ..., n$ then $a_{ij} = 1$ otherwise $a_{ij} = 0$.

Consider as an example matrix A shown in Fig. 1 which represents a network traffic. For this example, if each special card in ADMs controls one wavelength, then we need 21 cards. If some special cards are programmed to handle two consecutive wavelengths (this means that the bandpass number B is 2), then we would need 13 cards (8 for bandpasses, see Fig. 1.(a), 5 for single). If some special cards are programmed for three consecutive wavelengths (this means B is 3), then would we need 21 cards since there is no any bandpass when $B = 3$ as it can be seen in Fig. 1.(b).



(a) (b)

Fig. 1. (a) The Bandpasses when $B = 2$ (b) There is no any bandpass when $B = 3$.

Special cards in ADMs are expensive and IT managers try to reduce the number of these cards used in ADM. How is it possible? The answer is to reorder the wavelengths. For this purpose, the BP asks reordering of the rows of a given matrix so that the number of B non-overlapping consecutive 1's is maximized. Note that in the communication network, reordering of the rows of the matrix simply corresponds to a reassignment of the DWDM wavelengths.

Fig. 2 shows a reordering of the rows of the matrix given in Fig. 1. For this reordering, if some special cards are programmed for $B = 2$ consecutive wavelengths, then we need 13 cards again (8 for bandpasses, 5 for single). However, if some special cards are programmed for $B = 3$ consecutive wavelengths, then we need 13 cards (4 for bandpasses, 9 for single). Thus, this reordering of rows allows the use of fewer cards when $B = 3$.

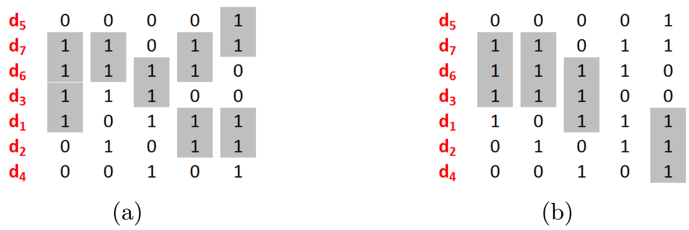


Fig. 2. (a) A reordering of the rows of the matrix in Fig. 1 and bandpasses when (a) $B = 2$ and (b) $B = 3$.

It is clear that to find an optimal solution to the BP, we must perform an exhaustive search over all row permutations. There are total of $m!$ different permutations of m wavelengths. This number grows faster than exponentially with m. Therefore, this is not reasonable. Babayev et al. also have proved that the BP is NP-hard [1].

Recent changes in ADM technology made the BP ineffective. We can explain the deficiencies of the BP as follows:

Technology allows an ADM to drop a wavelength even if it does not carry any information. Therefore, (a) a bandpass may contain zero elements. (b) The bandpass number B may be not fixed. (c) the BP ignores costs of the programmable cards. Thus, the BCP is proposed.

In the next section, we give a review of the mathematical models of the BCP. In Section 3, we introduce a library of problem instances for researchers to develop efficient computational solution methods, open for public use. In Section 4, we improve a simulated annealing (SA) metaheuristic to solve the BCP and also present computational results.

2 Mathematical Models of the BCP

In this section, we review all models of the BCP. We first introduce some notation that will be used throughout the models. Let $A = (a_{ij})$ be a binary matrix of dimension $m \times n$, $B_k = 2^k$ be a length of bands and c_k be a cost of the B_k-Band, where $(k = 0, 1, \ldots, t = \lfloor log_2 m \rfloor)$. π is a permutation of the rows such that $\pi = (\pi(1), \pi(2), ..., \pi(m))$. The goal of the BCP is to find an optimal permutation of rows of the matrix that minimizes the total cost of B_k-Bands in all columns subject to the constraints. The decision variables in the models are as follows:

$$x_{ir} = \begin{cases} 1 \text{ if row } i \text{ is relocated to position } r, \\ 0 \text{ otherwise,} \end{cases}$$

$$y_{ij}^k = \begin{cases} 1 \text{ if row } i \text{ is the first row of a } B_k\text{-Band in column } j, \\ 0 \text{ otherwise} \end{cases}$$

and

$$z_{ij}^k = \begin{cases} 1 \text{ if } a_{ij} \text{ is an element of a } B_k\text{-Band in column } j, \\ 0 \text{ otherwise.} \end{cases}$$

2.1 The Combinatorial Model

The combinatorial formulation of the BCP introduced first by Nuriyev et al. is the following [7].

$$\textit{Minimize} \sum_{k=0}^{t} \sum_{i=1}^{m-B_k+1} \sum_{j=1}^{n} c_k y_{\pi(i)j}^k \tag{1}$$

subject to

$$B_k \cdot y_{\pi(l)j}^k \leq \sum_{i=l}^{m} z_{\pi(i)j}^k, \; k = 0, ..., t, \; j = 1, ..., n, \; l = 1, ..., m - B_k + 1, \tag{2}$$

$$\sum_{k=0}^{t} \sum_{i=1}^{m-B_k+1} B_k \cdot y_{\pi(i)j}^k \geq \sum_{i=1}^{m} a_{ij}, \; j = 1, ..., n, \tag{3}$$

$$y_{\pi(l)j}^k + \sum_{i=l+1}^{l+B_k-1} \sum_{p=0}^{t} y_{\pi(i)j}^p \leq 1, \; k = 0, ..., t, \; j = 1, ..., n, \; l = 1, ..., m - B_k + 1, \tag{4}$$

$$\sum_{k=0}^{t} y_{\pi(i)j}^k \leq 1, \; i = 1, ..., m, \; j = 1, ..., n, \tag{5}$$

$$\sum_{k=0}^{t} z_{\pi(i)j}^k \geq a_{ij}, \; i = 1, ..., m, \; j = 1, ..., n, \tag{6}$$

$$\sum_{k=0}^{t} z_{\pi(i)j}^k \leq 1, \; i = 1, ..., m, \; j = 1, ..., n, \tag{7}$$

$$y_{ij}^k, z_{ij}^k \in \{0, 1\}, \; i = 1, ..., m, \; j = 1, ..., n, \; k = 0, ..., t, \tag{8}$$

The explanations of each constraint will be given at the end of this section.

2.2 The Linear Model

A linear programming model of the BCP introduced first by Gursoy et al. is the following [4].

$$\textit{Minimize} \sum_{k=0}^{t} \sum_{i=1}^{m-B_k+1} \sum_{j=1}^{n} c_k y_{ij}^k \tag{9}$$

subject to

$$\sum_{r=1}^{m} x_{ir} = 1, \; i = 1, \ldots, m \tag{10}$$

$$\sum_{i=1}^{m} x_{ir} = 1,\ r = 1, \ldots, m \tag{11}$$

$$B_k \cdot y_{lj}^k \leq \sum_{r=l}^{l+B_k-1} \sum_{i=1}^{m} a_{ij} \cdot x_{ir},\ k = 0, \ldots, t,\ j = 1, \ldots, n,\ l = 1, \ldots, m - B_k + 1 \tag{12}$$

$$\sum_{k=0}^{t} \sum_{i=1}^{m-B_k+1} B_k \cdot y_{ij}^k \geq \sum_{i=1}^{m} a_{ij},\ j = 1, \ldots n \tag{13}$$

$$\sum_{i=1}^{m-B_k+1} B_k \cdot y_{ij}^k = \sum_{i=1}^{m} z_{ij}^k,\ k = 0, \ldots, t,\ j = 1, \ldots, n \tag{14}$$

$$\sum_{i=l}^{l+B_k-1} y_{ij}^k \leq 1,\ k = 0, \ldots, t,\ j = 1, \ldots, n,\ l = 1, \ldots, m - B_k + 1 \tag{15}$$

$$\sum_{k=0}^{t} y_{ij}^k \leq 1,\ i = 1, \ldots, m,\ j = 1, \ldots, n \tag{16}$$

$$\sum_{k=0}^{t} z_{ij}^k \geq \sum_{r=1}^{m} a_{rj} \cdot x_{ri},\ i = 1, \ldots, m,\ j = 1, \ldots, n \tag{17}$$

$$\sum_{k=0}^{t} z_{ij}^k \leq 1,\ i = 1, \ldots, m,\ j = 1, \ldots, n \tag{18}$$

$$x_{ir},\ y_{ij}^k,\ z_{ij}^k \in \{0, 1\},\ i, r = 1, \ldots, m,\ j = 1, \ldots, n,\ k = 0, \ldots, t. \tag{19}$$

2.3 The Nonlinear Model

A nonlinear programming model of the BCP introduced first by Nuriyev et al. is the following [8].

$$\mathit{Minimize} \sum_{k=0}^{t} \sum_{i=1}^{m-B_k+1} \sum_{j=1}^{n} c_k y_{ij}^k \tag{20}$$

subject to

$$B_k \cdot y_{lj}^k \leq \sum_{r=l}^{l+B_k-1} \sum_{i=1}^{m} z_{ij}^k \cdot x_{ir},\ k = 0, \ldots, t,\ j = 1, \ldots, n,\ l = 1, \ldots, m - B_k + 1 \tag{21}$$

and (10),(11),(13),(15)-(19).

The constraints (10) express the fact that row i must be relocated into one new position r only, (11) express that only one row i must be relocated to each new position r, (2), (12) and (21) guarantee to find the coordinates of B_k-Bands, (13) say that the total length of bands in column j can not be less than the number of 1's in the same column, (14) identify the elements of B_k-Band in column j, (4), (5) and (15) guarantee that no two bands may have a common element, (3) and (16) guarantee that any non-zero entry of the permuted matrix belongs to a unique band B_k, (6), (7), (17) and (18) say that each non-zero entry of the permuted matrix has to be an element of a band B_k. In the models, all decision variables are binary which are indicated by (8) and (19).

The linear and nonlinear models have been implemented in GAMS (the General Algebraic Modeling System) and solved using the CPLEX and KNITRO solvers, respectively.

3 Online Library

As an extension of the Bandpass problem, the BCP is NP-hard. Therefore, it is needed to improve heuristic or metaheuristic algorithms. Potential researchers working on the BCP may need problem instances to check and compare their solutions.

First of all, we focused on what the comparison criteria of the solution algorithms would be. Main question was what the changing of the results was depending on? So, we considered all components of the BCP such as number of rows, number of columns, cost of a $B_k - Band$, density of the binary matrix.

Solution algorithms can be compared according to different number of rows, columns and different density of the binary matrix. Here, the density of a matrix is the ratio of the number of its nonzero entries to the total number of its entries.

It is clear that the upper bound of the length of a $B_k - Band$ depends on the number of rows. For example, if the number of rows equals 12, then there are $B_0 = 2^0 = 1$, $B_1 = 2^1 = 2$, $B_2 = 2^2 = 4$ and $B_3 = 2^3 = 8$ bands. The costs c_1, c_2, c_3 and c_4 are corresponding to each of these bands. Now, we face with a new question: How should we determine the costs? In particular, how should the price be increased from c_i to c_{i+1}? It cannot be random, since it is not suitable to compare algorithms. So, we identified six growth rates (ρ = 0.05, 0.1, 0.2, 0.3, 0.4 and 0.5) for this purpose, and used the following formula for all problem instances:

$c_{k+1} = (2 - \rho) \cdot c_k$, where $k = 0, 1, ..., t = \lfloor log_2 m \rfloor$

In the library, there are three different matrices of the same type. (TX-M1, TX-M2 and TX-M3), there are six different cost alternatives for the same matrix.

There are currently available 72 matrix types, 216 different matrices and 1296 problem instances. Potential contributors to this library can provide their instances. Information on new instances or optimum solutions for library problems is also appreciated (to be included into the library). At this point we should note that a solution must contain the permutation (ordering) of rows of the matrix, the starting position of each a $B_k - Band$ in all columns and the total cost.

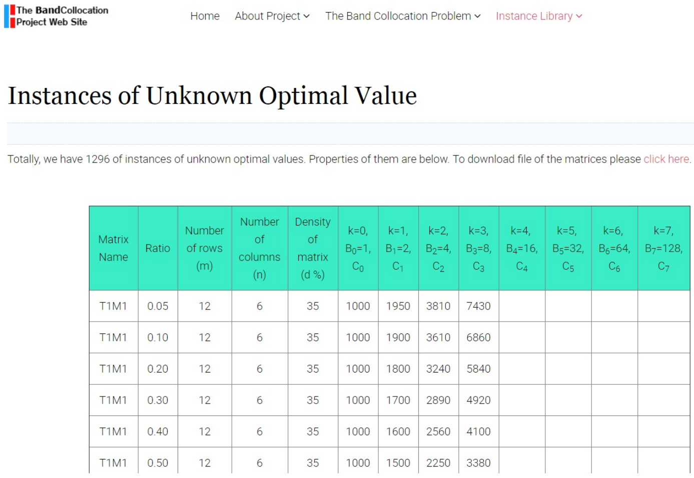
The Band Collocation Problem Library (BCPLib) is available online at http://www.izmir.edu.tr/bps. There are details about instances on "Remarks for Instances and Solutions" page. Fig. 3 shows the page of the Instances of Unknown Optimal Value which includes both the properties of 1296 instances and their excel file.

There will be properties of instances of known optimal value in "Instance of Known Optimal Value" and will be an excel file which will be included them. When we find some solutions to the instances, we will share results in "Solution for Instances of Unknown Optimal Value" and "Solutions for Instances of Known Optimal Value" pages.

Fig. 4 shows an excel file which includes instances. Each work sheet in the file has a TXMY matrix and six cost ratios.

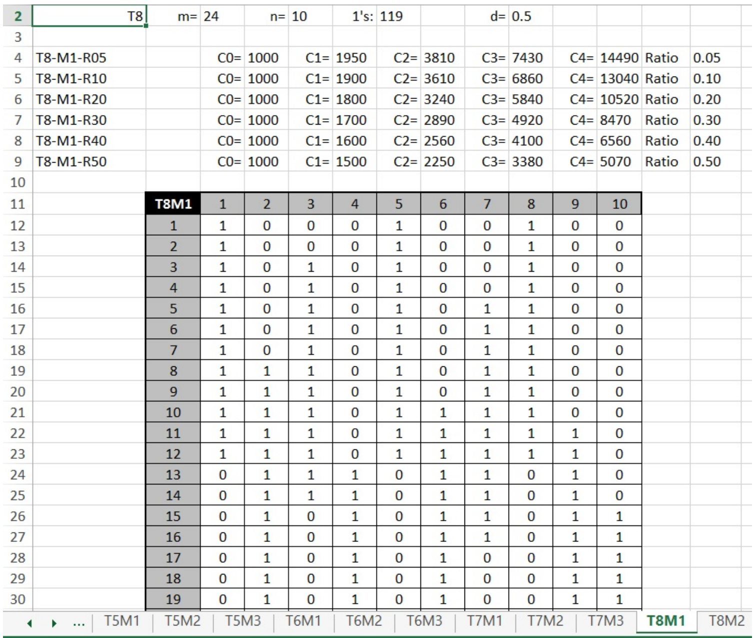
4 A Solution Approach to the BCP Using the Simulated Annealing

Simulated annealing (SA) is a popular metaheuristic approach for solving combinatorial optimization problems. The key point of simulated annealing is that it provides to find



Matrix Name	Ratio	Number of rows (m)	Number of columns (n)	Density of matrix (d %)	k=0, B_0=1, C_0	k=1, B_1=2, C_1	k=2, B_2=4, C_2	k=3, B_3=8, C_3	k=4, B_4=16, C_4	k=5, B_5=32, C_5	k=6, B_6=64, C_6	k=7, B_7=128, C_7
T1M1	0.05	12	6	35	1000	1950	3810	7430				
T1M1	0.10	12	6	35	1000	1900	3610	6860				
T1M1	0.20	12	6	35	1000	1800	3240	5840				
T1M1	0.30	12	6	35	1000	1700	2890	4920				
T1M1	0.40	12	6	35	1000	1600	2560	4100				
T1M1	0.50	12	6	35	1000	1500	2250	3380				

Fig. 3. Properties of the Instances.



2	T8	m= 24		n= 10		1's: 119		d= 0.5					
3													
4	T8-M1-R05		C0= 1000		C1= 1950		C2= 3810		C3= 7430		C4= 14490	Ratio	0.05
5	T8-M1-R10		C0= 1000		C1= 1900		C2= 3610		C3= 6860		C4= 13040	Ratio	0.10
6	T8-M1-R20		C0= 1000		C1= 1800		C2= 3240		C3= 5840		C4= 10520	Ratio	0.20
7	T8-M1-R30		C0= 1000		C1= 1700		C2= 2890		C3= 4920		C4= 8470	Ratio	0.30
8	T8-M1-R40		C0= 1000		C1= 1600		C2= 2560		C3= 4100		C4= 6560	Ratio	0.40
9	T8-M1-R50		C0= 1000		C1= 1500		C2= 2250		C3= 3380		C4= 5070	Ratio	0.50
10													

	T8M1	1	2	3	4	5	6	7	8	9	10
12	1	1	0	0	0	1	0	0	1	0	0
13	2	1	0	0	0	1	0	0	1	0	0
14	3	1	0	1	0	1	0	0	1	0	0
15	4	1	0	1	0	1	0	0	1	0	0
16	5	1	0	1	0	1	0	1	1	0	0
17	6	1	0	1	0	1	0	1	1	0	0
18	7	1	0	1	0	1	0	1	1	0	0
19	8	1	1	1	0	1	0	1	1	0	0
20	9	1	1	1	0	1	0	1	1	0	0
21	10	1	1	1	0	1	1	1	1	0	0
22	11	1	1	1	0	1	1	1	1	1	0
23	12	1	1	1	0	1	1	1	1	1	0
24	13	0	1	1	1	0	1	1	0	1	0
25	14	0	1	1	1	0	1	1	0	1	0
26	15	0	1	0	1	0	1	1	0	1	1
27	16	0	1	0	1	0	1	1	0	1	1
28	17	0	1	0	1	0	1	0	0	1	1
29	18	0	1	0	1	0	1	0	0	1	1
30	19	0	1	0	1	0	1	0	0	1	1

Fig. 4. The excel file of all instances.

a global optimum by escaping from local optima. SA introduced by Kirkpatrick et al. was inspired by the annealing process of physical systems [6]. At each iteration of a simulated annealing algorithm applied to an optimization problem, the objective function generates values for two solutions (the current solution and a newly produced solution) are compared. Better solutions are always accepted, while a fraction of bad solutions are accepted in the hope of escaping local optima in search of global optima. The probability of accepting bad solutions depends on a temperature parameter, which is typically geometrically decreasing with each iteration of the algorithm.

The pseudocode of the proposed SA algorithm is given in Algorithm 1. The proposed SA utilizes two reproduction operators, separately: one is to reverse a subset of the rows, the other is to swap two random rows. The first reproduction operator is actually $2-$*opt* local search method first proposed by Croes [2] for solving the traveling salesman problem. The second one is a well-known $\lambda - interchange$ mechanism introduced by Osman and Christofides [10]. In each reproduction operator, the initial temperature was set to 5000. The cooling parameter $\alpha = 0.9998$. The SA terminates when the temperature falls down 0.1 as indicated line 4 of Algorithm 1. The other possible termination condition is independent of the temperature and says that the simulated annealing can stop when there is no change in the cost of the matrix after a fixed number of iterations. Another termination condition is that the current cost value equals the optimum value. However, the optimum values of the test instances in BCPLib are not known, yet. Therefore, the SA algorithm lasts until the temperature falls down 0.1 which requires more time. The cost of the current solution (in line 5 of Algorithm 1) is computed by a dynamic programming algorithm proposed by Nuriyeva in [9].

Algorithm 1: Pseudocode of the SA.

1 Set the initial temperature T;
2 Set the cooling parameter α;
3 Generate an initial solution π and calculate its cost $f(\pi)$;
4 **while** $T > 0.1$ **do**
5   Generate a new solution π' by a reproduction operator and calculate its cost $f(\pi')$;
6     $\Delta f = f(\pi') - f(\pi)$;
7   **if** $\Delta f < 0$ **then**
8         $\pi = \pi'$;
9   **else if** $random[0,1] < \exp(-\frac{\Delta f}{T})$ **then**
10         $\pi = \pi'$;
11   **else**
12   Discard π';
13   **end**
15   `/* Update temperature */`
16     $T = T \times \alpha$;
17 **end**

4.1 Computational experiments

Our algorithm have been implemented in C and tested on i7-5600U machine with a 2.60 GHz processor and 8GB RAM. The test problems are taken from the BCPLib described above [11]. We perform 50 independent runs to get reliable statistical results. The entries of Table 1 and 2 presented below are:

Initial Cost	: the cost value of the input matrix,
m	: the number of rows in the matrix,
d	: the density of 1's in the matrix,
n	: the number of columns in the matrix,
ratio	: the cost ratio ρ,
Best Genetic	: the best value obtained by the genetic algorithm among 12 crossover and mutation techniques,
Best	: the best value obtained by the SA,
Worst	: the worst value obtained by the SA,
Average	: the average value among 50 runs,
Improvement	: the relative improvement between the best genetic value and our best value: $\frac{(\text{Best Genetic}-\text{Best})}{(\text{Best Genetic})} \times 100$,
Time	: average CPU time in seconds.

The computational results presented in Table 1 and 2 show that our proposed SA using both local search methods gives better results than the results obtained by the genetic algorithm reported in [11]. The genetic algorithm and our proposed SA found the same best values for T1-M1-R10, T1-M1-R30, T3-M1-R10. This is probably that these values are the optimum values for the instances. As it can be seen $\lambda - interchange$ local search method compared to $2 - opt$ is better in terms of the solution quality.

5 Conclusion

In this paper, we considered the Band Collocation Problem. We gave a review of its mathematical models. The linear and nonlinear models have been implemented in GAMS (the General Algebraic Modeling System) and solved using the CPLEX and KNITRO solvers, respectively. We also introduced the Band Collocation Problem Library (BCPLib) which is meant to provide researchers with a set of test problems having various properties. Generating problem instances whose optimal results known is the subject of future work. Finally, we improved a simulated annealing (SA) metaheuristic to solve the BCP. The proposed algorithm was performed using two local search methods for several test instances of the BCPLib and compared with the solutions obtained by a genetic algorithm. Experimental results showed that the proposed algorithm improves the quality of solutions. Moreover, $\lambda - interchange$ local search method is better than $2 - opt$ local search method for the BCP in the SA algorithm.

Acknowledgement

The authors would like to thank the anonymous referees for their valuable comments that considerably improved the presentation of the paper. This work is supported by the Scientific and Technological Re-search Council of Turkey-TUBITAK 3001 Project (Project No.:114F073)

References

1. Babayev, D. A., Bell, G. I. and Nuriyev, U. G.: The Bandpass Problem "Combinatorial Optimization and Library of Problems. J. Comb. Optim. 18, 151–172 (2009)
2. Croes, G. A.: A method for solving traveling salesman problems. Operations Res. 6, 791–812 (1958)
3. GAMS development Corporation (GDC), General Algebraic Modeling Systems (GAMS), Washington, DC. www.gams.com (2008)
4. Gursoy, A., Kutucu, H., Kurt, M., Nuriyev, U.: A Binary Integer Linear Programming Model for the Band Collocation Problem. Theoretical and Applied Aspects of Cybernetics, TAAC'2015, 131–134 (2015)
5. Kaminov, I.P. et al.: A Wideband All Optical WDM Network. IEEE Journal on Selected Areas in Communications 14 (5), 780–799 (1996)
6. Kirkpatrick, S., Gelatt, C.D. and Vecchi, M.P.: Optimization by Simulated Annealing. Science 220, 671 – 680 (1983)
7. Nuriyev, U., Kurt, M., Kutucu, H., Gursoy, A.: The Band Collocation Problem and Its Combinatorial Model. The International Conference Mathematical and Computational Modelling in Science and Technology, August 2-7, 140–141 (2015)
8. Nuriyev, U., Kutucu, H., Kurt, M. and Gursoy, A.:The band collocation problem in telecommunication networks. Proceedings of the 5th International Conference "Control and Optimization with Industrial Applications" (COIA-2015), 362–365 (2015)
9. Nuriyeva F.:On a generalized sequential partially covering problem. Appl. Comput. Math., 15(2) 240–243 (2016)

10. Osman, I.H. and Christofides, N.: Capacitated clustering problems by hybrid simulated annealing and tabu search. International Transactions in Operational Res., 1(3) 317 – 336 (1994)
11. The Band Collocation Problem Online Library (BCPLib), http://www.izmir.edu.tr/bps (Lad: 25.04.2016)

Table 1. Computational Results using $2 - opt$ local search method.

Instance	Initial Cost	m	n	d (%)	ratio	Best Genetic	Best	Worst	Average	improvement (%)	Time (sec.)
T1-M1-R10	24500	12	6	35	0.1	23140	23140	23140	23140	0.00	1.04
T1-M1-R30	23500	12	6	35	0.3	19660	19660	19660	19660	0.00	1.04
T3-M1-R10	50560	12	6	75	0.1	47680	47680	48330	47910	0.00	1.03
T3-M1-R30	41690	12	6	75	0.3	37340	36640	37640	37154	1.87	1.03
T4-M1-R10	43710	16	8	35	0.1	41960	41770	42060	41873	0.45	2.02
T4-M1-R30	41190	16	8	35	0.3	35710	35270	35920	35391	1.23	2.03
T6-M1-R10	89430	16	8	75	0.1	84820	84530	85730	84982	0.34	1.76
T6-M1-R30	67760	16	8	75	0.3	63780	63270	64440	63685	0.80	1.67
T7-M1-R10	78150	24	10	35	0.1	76700	75890	76800	76350	1.06	3.73
T7-M1-R30	69310	24	10	35	0.3	62950	62760	64760	63729	0.30	3.74
T9-M1-R10	161860	24	10	75	0.1	155870	154440	157650	155910	0.92	3.74
T9-M1-R30	125700	24	10	75	0.3	112050	110750	114640	113036	1.16	3.78
T10-M1-R10	123840	32	12	35	0.1	122780	120720	122630	121640	1.68	6.48
T10-M1-R30	105480	32	12	35	0.3	101370	96830	101400	99021	4.48	6.49
T12-M1-R10	258130	32	12	75	0.1	248400	247710	251640	249082	0.28	6.53
T12-M1-R30	172800	32	12	75	0.3	165360	163670	165670	164507	1.02	6.58
T13-M1-R10	185240	40	14	35	0.1	183170	179590	182350	181075	1.95	10.46
T13-M1-R30	162240	40	14	35	0.3	152800	147970	152100	149791	3.16	10.45
T15-M1-R10	377800	40	14	75	0.1	362300	360060	364960	362707	0.62	10.50
T15-M1-R30	262160	40	14	75	0.3	237600	236350	243210	239335	0.53	10.57
T16-M1-R10	254600	48	16	35	0.1	251980	247600	250990	249478	1.74	15.23
T16-M1-R30	222500	48	16	35	0.3	205570	199890	208520	204040	2.76	15.24
T18-M1-R10	510780	48	16	75	0.1	493430	489480	496280	493317	0.80	15.27
T18-M1-R30	350030	48	16	75	0.3	326330	324820	332620	328721	0.46	15.40
T19-M1-R10	340640	56	18	35	0.1	328690	324820	327460	325948	1.18	20.74
T19-M1-R30	314490	56	18	35	0.3	276330	268590	278200	273353	2.80	20.76
T21-M1-R10	686690	56	18	75	0.1	646570	645950	656520	650793	0.10	20.84
T21-M1-R30	472820	56	18	75	0.3	424070	420690	437540	431766	0.80	20.98
T22-M1-R10	425490	64	20	35	0.1	411140	405340	411210	407808	1.41	27.18
T22-M1-R30	375220	64	20	35	0.3	327420	315620	329530	323386	3.60	27.18
T24-M1-R10	859860	64	20	75	0.1	825270	824920	832590	827887	0.04	27.31
T24-M1-R30	489600	64	20	75	0.3	484440	481980	489600	486262	0.51	27.49
T25-M1-R10	527560	72	22	35	0.1	517920	509940	514550	511843	1.54	35.12
T25-M1-R30	462480	72	22	35	0.3	426750	411170	429560	420558	3.65	35.31
T27-M1-R10	1065400	72	22	75	0.1	1017110	1011590	1025480	1018535	0.54	35.43
T27-M1-R30	624480	72	22	75	0.3	599250	592350	604780	598669	1.15	35.58
T28-M1-R10	641050	80	24	35	0.1	622670	607340	616550	612774	2.46	44.12
T28-M1-R30	568120	80	24	35	0.3	493910	480370	502370	489237	2.74	44.25
T30-M1-R10	1314500	80	24	75	0.1	1206320	1197910	1217230	1206502	0.70	44.54
T30-M1-R30	777680	80	24	75	0.3	703520	695570	714690	704054	1.13	45.36
T31-M1-R10	767110	88	26	35	0.1	737840	719090	728970	723282	2.54	60.07
T31-M1-R30	680630	88	26	35	0.3	565920	539520	561860	549678	4.66	54.63
T33-M1-R10	1514390	88	26	75	0.1	1446220	1434670	1460990	1448456	0.80	54.69
T33-M1-R30	921580	88	26	75	0.3	861830	858160	879990	869746	0.43	56.27
T34-M1-R10	892920	96	28	35	0.1	881670	868240	875100	871299	1.52	65.91
T34-M1-R30	778400	96	28	35	0.3	716030	704370	726510	714474	1.63	65.69
T36-M1-R10	1792900	96	28	75	0.1	1730450	1723880	1750670	1738069	0.38	66.30
T36-M1-R30	1069280	96	28	75	0.3	1040420	1029500	1053770	1042701	1.05	66.61
T37-M1-R10	225600	56	12	35	0.1	216180	211830	214820	213401	2.01	14.00
T37-M1-R30	207400	56	12	35	0.3	177850	168990	176960	173282	4.98	14.03
T39-M1-R10	457780	56	12	75	0.1	426140	422620	432070	426282	0.83	14.04
T39-M1-R30	315410	56	12	75	0.3	278760	276150	286270	280926	0.94	14.17
T40-M1-R10	253890	64	12	35	0.1	241010	236840	240670	239164	1.73	16.51
T40-M1-R30	220980	64	12	35	0.3	190230	181710	189890	185774	4.48	16.50
T42-M1-R10	517620	64	12	75	0.1	491910	486020	494450	490491	1.20	16.55
T42-M1-R30	293760	64	12	75	0.3	292150	290300	293760	293402	0.63	16.68
T43-M1-R10	287590	72	12	35	0.1	278070	269870	274070	271595	2.95	19.82
T43-M1-R30	252870	72	12	35	0.3	220250	207960	219100	212597	5.58	19.65
T45-M1-R10	576240	72	12	75	0.1	547710	539530	548340	543795	1.49	19.56
T45-M1-R30	341250	72	12	75	0.3	321260	316430	325900	320419	1.50	20.53
T46-M1-R10	318760	80	12	35	0.1	305810	296210	301170	298161	3.14	24.15
T46-M1-R30	276330	80	12	35	0.3	232970	219240	231080	224844	5.89	27.14
T48-M1-R10	656320	80	12	75	0.1	600090	591300	605790	595983	1.46	23.99
T48-M1-R30	392230	80	12	75	0.3	349090	345570	359660	352293	1.01	22.38
T49-M1-R10	354240	88	12	35	0.1	336450	321850	329660	325968	4.34	24.95
T49-M1-R30	316900	88	12	35	0.3	248310	237010	249990	242347	4.55	25.03
T51-M1-R10	703220	88	12	75	0.1	659660	654950	670830	661264	0.71	25.16
T51-M1-R30	420250	88	12	75	0.3	386880	385770	399620	393017	0.29	25.35
T52-M1-R10	382010	96	12	35	0.1	371540	359100	364510	361886	3.35	27.84
T52-M1-R30	334090	96	12	35	0.3	287280	274980	288230	280537	4.28	27.87
T54-M1-R10	763250	96	12	75	0.1	718540	714850	731960	721593	0.51	28.05
T54-M1-R30	454580	96	12	75	0.3	424010	422410	437350	428571	0.38	28.22

Table 2. Computational Results using $\lambda - interchange$ local search method.

Instance	Initial Cost	m	n	d (%)	ratio	Best Genetic	Best	Worst	Average	improvement (%)	Time (sec.)
T1-M1-R10	24500	12	6	35	0.1	23140	23140	23330	23144	0.00	0.85
T1-M1-R30	23500	12	6	35	0.3	19660	19660	20170	19670	0.00	0.85
T3-M1-R10	50560	12	6	75	0.1	47680	47680	48420	47936	0.00	0.86
T3-M1-R30	41690	12	6	75	0.3	37340	36640	38310	37136	1.87	0.85
T4-M1-R10	43710	16	8	35	0.1	41960	41860	42340	42041	0.24	1.68
T4-M1-R30	41190	16	8	35	0.3	35710	35570	37050	36109	0.39	1.69
T6-M1-R10	89430	16	8	75	0.1	84820	84530	85440	84939	0.34	1.72
T6-M1-R30	67760	16	8	75	0.3	63780	63270	64290	63827	0.80	1.69
T7-M1-R10	78150	24	10	35	0.1	76700	76070	77440	76710	0.82	3.83
T7-M1-R30	69310	24	10	35	0.3	62950	62790	65590	63983	0.25	3.84
T9-M1-R10	161860	24	10	75	0.1	155870	154410	156880	155409	0.94	3.97
T9-M1-R30	125700	24	10	75	0.3	112050	110750	113640	112506	1.16	3.93
T10-M1-R10	123840	32	12	35	0.1	122780	121620	123610	122594	0.94	6.76
T10-M1-R30	105480	32	12	35	0.3	101370	96330	102580	99527	4.97	6.69
T12-M1-R10	258130	32	12	75	0.1	248400	246040	248960	247800	0.95	6.97
T12-M1-R30	172800	32	12	75	0.3	165360	163170	165230	164364	1.32	6.68
T13-M1-R10	185240	40	14	35	0.1	183170	180790	183800	182052	1.30	10.61
T13-M1-R30	162240	40	14	35	0.3	152800	147690	152330	150137	3.34	10.62
T15-M1-R10	377800	40	14	75	0.1	362300	358140	362650	360460	1.15	10.68
T15-M1-R30	262160	40	14	75	0.3	237600	235260	239530	237175	0.98	10.73
T16-M1-R10	254600	48	16	35	0.1	251980	248540	251650	250212	1.37	15.46
T16-M1-R30	222500	48	16	35	0.3	205570	198160	206850	202997	3.60	15.48
T18-M1-R10	510780	48	16	75	0.1	493430	486900	493000	489605	1.32	15.94
T18-M1-R30	350030	48	16	75	0.3	326330	319100	327000	323881	2.22	15.97
T19-M1-R10	340640	56	18	35	0.1	328690	325660	330010	327338	0.92	21.53
T19-M1-R30	314490	56	18	35	0.3	276330	267780	277690	271147	3.09	21.51
T21-M1-R10	686690	56	18	75	0.1	646570	641830	648540	644890	0.73	21.64
T21-M1-R30	472820	56	18	75	0.3	424070	416190	425330	420458	1.86	21.76
T22-M1-R10	425490	64	20	35	0.1	411140	406160	412610	409402	1.21	28.17
T22-M1-R30	375220	64	20	35	0.3	327420	312560	328440	320969	4.54	28.24
T24-M1-R10	859860	64	20	75	0.1	825270	814710	824910	819750	1.28	28.96
T24-M1-R30	489600	64	20	75	0.3	484440	480280	489600	486612	0.86	28.55
T25-M1-R10	527560	72	22	35	0.1	517920	510110	515430	513040	1.51	35.38
T25-M1-R30	462480	72	22	35	0.3	426750	408930	420230	414461	4.18	35.74
T27-M1-R10	1065400	72	22	75	0.1	1017110	1005880	1014550	1009427	1.10	39.02
T27-M1-R30	624480	72	22	75	0.3	599250	590470	598910	595666	1.47	44.02
T28-M1-R10	641050	80	24	35	0.1	622670	610900	616330	613781	1.89	54.54
T28-M1-R30	568120	80	24	35	0.3	493910	470530	488900	479258	4.73	50.56
T30-M1-R10	1314500	80	24	75	0.1	1206320	1189580	1201650	1194789	1.39	44.56
T30-M1-R30	777680	80	24	75	0.3	703520	684100	697060	691123	2.76	44.84
T31-M1-R10	767110	88	26	35	0.1	737840	719550	733890	726716	2.48	54.23
T31-M1-R30	680630	88	26	35	0.3	565920	529220	549270	537201	6.49	54.57
T33-M1-R10	1514390	88	26	75	0.1	1446220	1422480	1432650	1428080	1.64	55.37
T33-M1-R30	921580	88	26	75	0.3	861830	840680	858170	848478	2.45	56.28
T34-M1-R10	892920	96	28	35	0.1	881670	868940	875870	872717	1.44	66.67
T34-M1-R30	778400	96	28	35	0.3	716030	688390	708550	696284	3.86	66.68
T36-M1-R10	1792900	96	28	75	0.1	1730450	1703600	1723180	1712658	1.55	67.04
T36-M1-R30	1069280	96	28	75	0.3	1040420	1014750	1025700	1020757	2.47	67.45
T37-M1-R10	225600	56	12	35	0.1	216180	213450	216320	214827	1.26	14.14
T37-M1-R30	207400	56	12	35	0.3	177850	168770	175100	171903	5.11	14.15
T39-M1-R10	457780	56	12	75	0.1	426140	418630	426090	422009	1.76	14.23
T39-M1-R30	315410	56	12	75	0.3	278760	268530	276790	272117	3.67	14.31
T40-M1-R10	253890	64	12	35	0.1	241010	238630	242690	240730	0.99	16.68
T40-M1-R30	220980	64	12	35	0.3	190230	179560	187270	183624	5.61	16.70
T42-M1-R10	517620	64	12	75	0.1	491910	481710	488650	484858	2.07	16.78
T42-M1-R30	293760	64	12	75	0.3	292150	288690	293760	293520	1.18	16.89
T43-M1-R10	287590	72	12	35	0.1	278070	271020	275350	273560	2.54	19.66
T43-M1-R30	252870	72	12	35	0.3	220250	204450	213230	209032	7.17	19.68
T45-M1-R10	576240	72	12	75	0.1	547710	534920	541480	537892	2.34	19.79
T45-M1-R30	341250	72	12	75	0.3	321260	315680	320820	318202	1.74	19.90
T46-M1-R10	318760	80	12	35	0.1	305810	298190	302830	300189	2.49	22.60
T46-M1-R30	276330	80	12	35	0.3	232970	213650	223490	219515	8.29	22.65
T48-M1-R10	656320	80	12	75	0.1	600090	584150	592600	588024	2.66	22.77
T48-M1-R30	392230	80	12	75	0.3	349090	339870	347420	344442	2.64	22.94
T49-M1-R10	354240	88	12	35	0.1	336450	325350	333560	329375	3.30	25.57
T49-M1-R30	316900	88	12	35	0.3	248310	230280	242380	236380	7.26	26.69
T51-M1-R10	703220	88	12	75	0.1	659660	644100	652570	648823	2.36	31.10
T51-M1-R30	420250	88	12	75	0.3	386880	375280	385330	379620	3.00	31.27
T52-M1-R10	382010	96	12	35	0.1	371540	361130	367440	364329	2.80	34.43
T52-M1-R30	334090	96	12	35	0.3	287280	265990	276520	271182	7.41	34.49
T54-M1-R10	763250	96	12	75	0.1	718540	702880	711630	707132	2.18	34.69
T54-M1-R30	454580	96	12	75	0.3	424010	408520	420580	415028	3.65	34.90