

## APPLICATION OF ANT COLONY OPTIMIZATION METAHEURISTIC ON SET COVERING PROBLEMS

CHRISTIAN ALVIN H. BUHAT, JERSON KEN L. VILLAMIN, AND GENARO A. CUARESMA

**ABSTRACT.** Ant Colony Optimization (ACO) metaheuristic is a multi-agent system in which the behaviour of each ant is inspired by the foraging behaviour of real ants to solve optimization problem. Set Covering Problems (SCP), on the other hand, deal with maximizing the coverage of every subset while the weight nodes used must be minimized. In this paper, ACO was adapted and used to solve a case of Set Covering Problem. The adapted ACO for solving the SCP was implemented as a computer program using SciLab 5.4.1. The problem of determining the optimal location of Wi-Fi Access Points using the 802.11n protocol in the UP Los Banos Math Building (MB) was solved using this metaheuristic. Results show that in order to have 100% coverage of the MB, 7 access points are required. Methodology of the study can be adapted and results of the study can be used by decision makers on related optimization problems.

### 1. INTRODUCTION

The Set Covering Problem (SCP) is a typical optimization problem that serves as a model for many applications in the real world. It is a Combinatorial Optimization Problem (COP) which can be formulated as an Integer Linear Programming (ILP) where the goal is to minimize the number of sets such that every element of the set from the universe will be covered and every set is either in the set cover or not. SCP has been used to model problems such as vehicle routing, nurse scheduling, airline crew scheduling, facility location problem, and resource allocation problem (Schiff, 2013).

According to Lessing et al (2004), SCP is an NP-hard problem with many developed algorithms that can be used to solve it. Some algorithms can solve a limited size and is also time consuming that is why metaheuristic is advised for solving such problem.

Metaheuristic is used to define heuristic methods and it is also a set of algorithmic concepts applicable to a very large set of problems. Ren, et al. (2008), reported that list of kind of heuristics were applied for the SCP and these are Genetic Algorithm (GA), Simulated Annealing Algorithm, and Tabu Search Algorithm. They also conducted a research to solve SCP based from the recently developed, population-based metaheuristic named Ant Colony Optimization (ACO) algorithm.

ACO was published in the early 90's and it is a probabilistic technique about the simulation of real behavior of ants regarding how they function through indirect communication via pheromones (Brezina and Cickova, 2011). Pheromones are chemical substances excreted by ants to attract other ants that are seeking for food. Biologists have observed that ants tend to find the path with the shortest distance between a source of food and their colony which leads in the conclusion that it would be useful to study their individual behavior (Buezas. 2010). Solutions to a given optimization problem are obtained

---

Received by the editors 26 May 2021; accepted 11 January 2022; published online 20, January 2022.

2020 *Mathematics Subject Classification.* Primary 90B50, 90B80; Secondary 90B10, 90B90.

*Key words and phrases.* Constrained optimization, decision theory, information technology, optimal placement, metaheuristics.

by a set of software agents called artificial ants. In applying ACO, the optimization problem must be transformed into the problem of finding the best path on a weighted graph. Hereafter, ants incrementally gets solutions as the graph goes on.

ACO is commonly used in solving Traveling Salesman Problem (TSP), but it is also applicable on Set Covering Problems (Dorigo and Stutzle, 2004).

Given a set of facilities and the areas covered by the facility, a number of ants determine what facility to choose using a probability function for ACO which is based on pheromone trails and heuristic information, while the area it chooses is randomized. After the chosen areas by the ant satisfies the system constraints or covers all the areas in the system, a next ant will proceed, and the pheromone trails and heuristic information will be updated. This will continue until all ants have passed, and the best solution for the problem is returned and recorded.

On 2008, Ren et al proposed a novel ACO-based approach, called Ant-Cover. It has three main differences between other existing ACO-based approaches: First, for constructing solutions, the approach used single-row-oriented method, next, the use of LaGrange dual information for the search in the solution space, lastly, the use of local search procedure is developed to maintain the feasibility and for the solutions to be improved by ants. This study used a customized Ant-Cover to get a faster near-optimal solution than other metaheuristics.

Due to the numerous problems about set covering, this paper is to adapt the Ant Colony Optimization algorithm to solve a Set Cover Problem. In line, this paper is concerned in determining the optimal placement of Wi-Fi Access Points in the University of the Philippines Los Baños (UPLB) Math Building using ACO metaheuristic.

## 2. THEORETICAL FRAMEWORK

The Set Covering Problem is a Binary Integer Programming model that seeks to find the minimum number of facilities to be installed and their locations to each demand node is covered by at least one facility (Torregas, 1970).

Let  $I$  = set of all possible locations for installing a facility,

$J$  = set of all areas which are to be covered,

$$x_i = \begin{cases} 1, & \text{if a facility is to be installed at site } i, \\ 0, & \text{otherwise;} \end{cases}$$

$$y_j = \begin{cases} 1, & \text{if whenever a facility is installed at site } i \text{ the facility covers area } j, \\ 0, & \text{otherwise.} \end{cases}$$

The model can be constructed as:

$$\text{minimize } z = \sum_{i \in I} x_i, \tag{2.1}$$

subject to

$$\sum_{i \in I} x_i y_{ij} \geq 1, \forall j \in J; \tag{2.2}$$

$$x_i = \{0, 1\}, \forall i \in I. \tag{2.3}$$

The objective function (1) determines the goal of the problem which is to minimize the number of facilities used. The first constraint (2) satisfies the condition that each area  $j$  should be covered by at least one facility and the second constraint (3) serves as the binary constraint.

### 3. RESULTS AND DISCUSSION

#### Pseudocode

This paper adapts a similar pseudocode to Ren et al. (2008) which uses ACO on solving SCP. The following is the procedure of adapting ACO to solving SCP.

```

Pre-process the given SCP and initialize results
Initialize pheromone trails, heuristic information, and related parameters
while termination condition is not met do
for ant i=1 to n (number of ants) do
(a)Construct a solution based on SROM
if system constraints are satisfied then move to next ant
else repeat (a)
end for
Update pheromone trails and heuristic information
end while
Return the best solution found

```

#### Preprocessing of SCP (General)

Based on the SCP Formulation, we transform the problem into an  $m \times n$  matrix, with  $m$  being the areas to be covered and  $n$  being the facilities. An element of the matrix will have a value of 1 defining that that area has been covered under that facility, while 0 if it has yet to be covered, i.e.,

$$M(i,j) = \begin{cases} 1, & \text{if facility } j \text{ covers area } i, \\ 0, & \text{otherwise.} \end{cases}$$

TABLE 1. Example of a preprocessed SCP Model

	1	2	3	4	5	6	.	.	.	.	$n$
1	1	0	0	1	0	1	0	0	0	0	1
2	0	1	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	1	0	0
4	0	0	0	1	0	0	0	0	0	0	1
5	0	0	0	0	0	1	0	0	0	0	0
6	0	1	0	0	0	0	0	0	0	0	0
.	0	0	0	0	0	0	0	1	0	0	0
.	1	0	0	0	1	0	0	0	0	0	0
.	0	0	0	0	0	0	1	0	0	0	0
.	0	0	0	0	0	0	0	0	0	1	0
$m$	1	0	0	0	0	0	0	0	0	0	0

In the above matrix, this means that area 1 can be covered by facilities in locations 1, 4, 6, ...,  $n$ .

#### Parameters Used

Values of parameters used in metaheuristics such as ACO are determined by the proponent where a slight change in such values can generate a large difference in the results. In this paper we used values suggested by Ren and others (2008) for the value of our  $\beta$  and  $\rho$ . Meanwhile, values for number of

iterations, initial pheromone trails and heuristic information are standard values used in ACO. Values and definition of variables are given in the table below.

TABLE 2. Definitions and Values of Parameters Used

Variable	Definition	Value	Reference
$\beta$	Importance of Heuristic Factor to Pheromone	5	Ren and others (2008)
$\rho$	Pheromone Persistence	0.99	Ren and others (2008)
ants	Number of iterations	1000	Assumed
tau(1)	Initial Pheromone Trail	1	Assumed
heu(1)	Initial Heuristic Information	1	Assumed

Single row-oriented solution construction method is a novel single-row-oriented solution construction method which can generate solutions in a faster way (compared to other branch-and-bound/branch-and-cut algorithms) and allows ants to explore broader solution areas (Ren and others, 2008).

Two of the main ideas of SROM on how it adapts ACO in solving SCP are to:

- Randomly select a row  $I$  from the row set  $I$
- Select a column from the column set  $J$  using the probability
- Randomly select a row  $i$  from the row set  $I$
- Select a column from the column set  $J$  using the probability

$$P(s_t = j | r_t = i, i \in R_{t-1}) = \begin{cases} \frac{\tau_j \eta_j^\beta}{\sum_{q \in J_i} \tau_q \eta_q^\beta} & \text{if } j \in J, \\ 0 & \text{otherwise,} \end{cases}$$

where  $R_{t-1}$  denotes the set of uncovered rows before step  $t$ , and  $r_t$  denotes the row chosen at step  $t$ . The time complexity of selecting a column from the set  $J_i$  is  $O(|J_i|)$ . Let  $d$  be the density of SCP, the average number of columns covering a row can be obtained,  $\tilde{n} = dn$ . Then we can get a complete solution  $S$  with  $k$  columns in  $O(k\tilde{n})$  time. This time complexity is linear with the value of  $\tilde{n}$ . Generally,  $\tilde{n} \ll n$  holds (Ren and others, 2008).

### Pheromone Trail Update

Pheromone trails are decreased uniformly in order to simulate evaporation and allow ants to forget part of the history experience. Ants then deposit an amount of pheromone on the column contained in the best solution. After all the ants have completed a solution, the trails are updated as follows:

$$\tau_j \leftarrow \rho \tau_j + \Delta \tau_j, \quad \forall j \in J,$$

$$\Delta \tau_j = \begin{cases} \frac{1}{\sum_{q \in S_{gb}} C_q} & \text{if } j \in S_{gb}, \\ 0 & \text{otherwise.} \end{cases}$$

Here,

$$\begin{aligned} \rho &= \text{pheromone persistence } (0 \leq \rho < 1), \\ \Delta \tau_j &= \text{amount of pheromone put on column } j, \\ \sum_{q \in S_{gb}} C_q &= \text{total amount of cost.} \end{aligned}$$

Pheromone trail updates help the model to adapt to real-time changes, which is one of its advantages over other heuristics such as simulated annealing and genetic algorithm (Selvi and Umarani, 2010).

### Heuristic Information Update

Heuristic Information considers the dual information associated with the still uncovered rows. It mines the specific information of the SCP and provides a good guideline for ants' search in the solution space. It is updated through:

$$\varphi_j = |I_j \cap R|,$$

$$\eta_j = \frac{\varphi_j}{c_j},$$

where:

- $\varphi_j$  = number of new rows that can be covered by column  $j$ ,
- $R$  = set of still uncovered rows.

These updates help the considered artificial ants in the heuristic to obtain the optimal solution that we need for our LP model (Abd-Alsabour and others, 2013).

### Program Description

Application of Ant Colony Optimization Metaheuristic on Set Covering Problem program generates a near-optimal solution to a preprocessed SCP using ACO Metaheuristic. ACO metaheuristic was implemented in SciLab 5.4.1 under Microsoft Windows 7 on Intel Pentium Dual CPU 2.16 GHz, 2.0 Gb RAM with parameter values,  $\beta = 5$ ,  $\rho = 0.99$ , ants=1000, tau(1)=heu(1)=1. The user inputs a specified number of facilities and areas associated, or provides a preprocessed  $m \times n$  matrix which will then be run to provide a near-optimal value, with its associated optimal solution, number of iterations needed before attaining the near-optimal value, and the running time of the program. Table 3 summarizes the description of the ACO on SCP program.

TABLE 3. Program Description of ACO on SCP

PROGRAM TITLE	Application of Ant Colony Optimization Metaheuristic on Set Covering Problems
DESCRIPTION	This program solves for a near- optimal Solution of a Set Covering Problem by applying ACO Metaheuristic
PROGRAMMING LANGUAGE	Scilab 5.4.1
INPUT	Number of facilities and Areas associated/ m x n matrix (preprocessed SCP)
OUTPUT	Optimal Value of the problem, Associated Optimal Solution, Number of Iterations, Minimum Running Time

## 4. APPLICATION

In order to further determine the effectiveness and efficiency of how ACO adapts in solving ACO, the study was applied to a local set covering problem and used to determine the optimal place of Wi-Fi Routers in Math Building. Data used was provided by the UPLB-ITC and OVCPD. In this paper, 802.11n Wi-Fi Routers were considered as the facility, while area was based on the division of area/rooms in Math Building.

Since math building is a two-storey building, the problem was divided into two parts (1st floor and 2nd floor) for better explanation of the study and application. Note however that the problem can also be formulated as one SCP Model. Constraints of the model indicate that an area should be covered by at least one repeater. This will guarantee that the entire Math Building is covered.

As in Section 2, the related variables are defined as below in the context of WiFi routers now:

Let  $I$  = set of all possible locations for installing a Wi-Fi router

$J$  = set of all rooms/areas which are to be covered

$$x_i = \begin{cases} 1, & \text{if a Wi-Fi router is to be installed at room } i \\ 0, & \text{otherwise} \end{cases}$$

$$y_j = \begin{cases} 1, & \text{if whenever a Wi-Fi router is installed at room } i \text{ the facility covers room } j \\ 0, & \text{otherwise} \end{cases}$$

Figure 1 serve as a map where  $x_i$ 's are the areas in Math Building being covered,  $i$  being the area number, as shown:

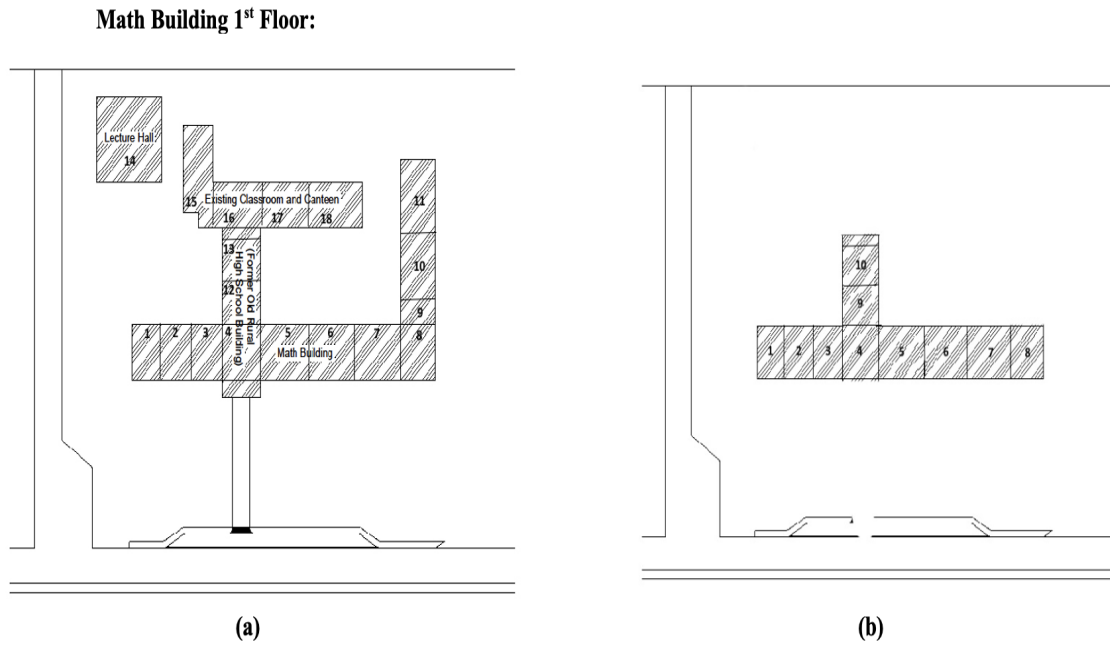


FIGURE 1. Old Math Building 1st (a) and 2nd (b) Floor Plan Generated through AutoCAD

**Math Building First Floor:**

$$\text{Minimize } z = \sum_{i=1}^{18} x_i$$

subject to the constraints

$$\begin{aligned}
x_1 + x_2 + x_3 &\geq 1 \\
x_1 + x_2 + x_3 + x_4 + x_{12} &\geq 1 \\
x_1 + x_2 + x_3 + x_4 + x_5 + x_{12} + x_{13} &\geq 1 \\
x_2 + x_3 + x_4 + x_5 + x_{12} + x_{13} &\geq 1 \\
x_3 + x_4 + x_5 + x_6 + x_{12} + x_{13} &\geq 1 \\
x_5 + x_6 + x_7 &\geq 1 \\
x_6 + x_7 + x_8 + x_9 + x_{10} &\geq 1 \\
x_6 + x_7 + x_8 + x_9 + x_{10} &\geq 1 \\
x_7 + x_8 + x_9 + x_{10} &\geq 1 \\
x_7 + x_8 + x_9 + x_{10} + x_{11} &\geq 1 \\
x_9 + x_{10} + x_{11} + x_{18} &\geq 1 \\
x_3 + x_4 + x_{12} + x_{13} + x_{16} &\geq 1 \\
x_3 + x_4 + x_5 + x_{12} + x_{13} + x_{16} + x_{17} &\geq 1 \\
x_{14} + x_{15} &\geq 1 \\
x_{12} + x_{13} + x_{15} + x_{16} &\geq 1 \\
x_{12} + x_{13} + x_{16} + x_{17} + x_{18} &\geq 1 \\
x_{10} + x_{11} + x_{18} &\geq 1 \\
x_i &= \{0, 1\}, i = 1, 2, 3, \dots, 18
\end{aligned}$$

**Math Building Second Floor:**

$$\text{Minimize } z = \sum_{i=1}^{10} x_i$$

subject to

$$\begin{aligned}
x_1 + x_2 + x_3 &\geq 1 \\
x_1 + x_2 + x_3 + x_4 + x_9 &\geq 1 \\
x_1 + x_2 + x_3 + x_4 + x_5 + x_9 + x_{10} &\geq 1 \\
x_2 + x_3 + x_4 + x_5 + x_9 + x_{10} &\geq 1 \\
x_3 + x_4 + x_5 + x_6 + x_9 + x_{10} &\geq 1 \\
x_4 + x_5 + x_6 + x_7 + x_9 &\geq 1 \\
x_5 + x_6 + x_7 + x_8 &\geq 1 \\
x_6 + x_7 + x_8 &\geq 1 \\
x_2 + x_3 + x_4 + x_5 + x_9 + x_{10} &\geq 1 \\
x_3 + x_4 + x_5 + x_9 + x_{10} &\geq 1 \\
x_i &= \{0, 1\}, i = 1, 2, 3, \dots, 10
\end{aligned}$$

**Preprocessing of SCP (Specific)**

Using the definition of the decision variables and a range of 140ft., with  $x_i$ ,  $i=1,\dots,18$  for the first floor and  $j=1,\dots,10$  for the second floor, the pre-processed matrix for the SCP is given in Table 4 and 5 for the first and second floors respectively in the Math Building.

**Math Building 1st Floor:**

TABLE 4.  $m \times n$  matrix representing SCP Formulation of MB First Floor

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	1	1	1	1	1	0	0	0	0	0	0	1	1	0	0	0	0	0
4	0	1	1	1	1	0	0	0	0	0	0	1	1	0	0	0	0	0
5	0	0	1	1	1	1	0	0	0	0	0	0	1	0	0	0	0	0
6	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0
10	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	1
11	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1
12	0	1	1	1	1	0	0	0	0	0	0	1	1	0	1	1	1	0
13	0	0	1	1	1	0	0	0	0	0	0	1	1	0	1	1	1	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0
16	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	1	0
17	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0
18	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1

**Math Building Second Floor:**

TABLE 5.  $m \times n$  matrix representing SCP Formulation of MB Second Floor

	1	2	3	4	5	6	7	8	9	10
1	1	1	1	0	0	0	0	0	0	0
2	1	1	1	1	0	0	0	0	1	0
3	1	1	1	1	1	0	0	0	1	1
4	0	1	1	1	1	1	0	0	1	1
5	0	0	1	1	1	1	1	0	1	1
6	0	0	0	0	1	1	1	1	0	0
7	0	0	0	0	0	1	1	1	0	0
8	0	0	0	0	0	0	1	1	0	0
9	0	1	1	1	1	1	0	0	1	1
10	0	0	1	1	1	0	0	0	1	1



**Program Results**

After 1000 iterations/ants, ACO on SCP Program determined a near-optimal solution of 7 for the UPLB Math Building (5 for the first floor, 2 for the second floor). These Wi-Fi routers must then be placed at area number: 3, 6, 10, 12, 14 for the first floor, and 3, 6 for the second floor. However, the associated optimal solution given by the program may not be unique and may have other results which still give a near-optimal solution to the problem.

**Optimal Wi-Fi Routers:**

1st Floor: 5  
 2nd Floor: 2

**Associated Optimal Solution:**

1st Floor: 0-0-1-0-0-1-0-0-0-1-0-1-0-1-0-0-0-0  
 2nd Floor: 0-0-1-0-0-1-0-0-0-0

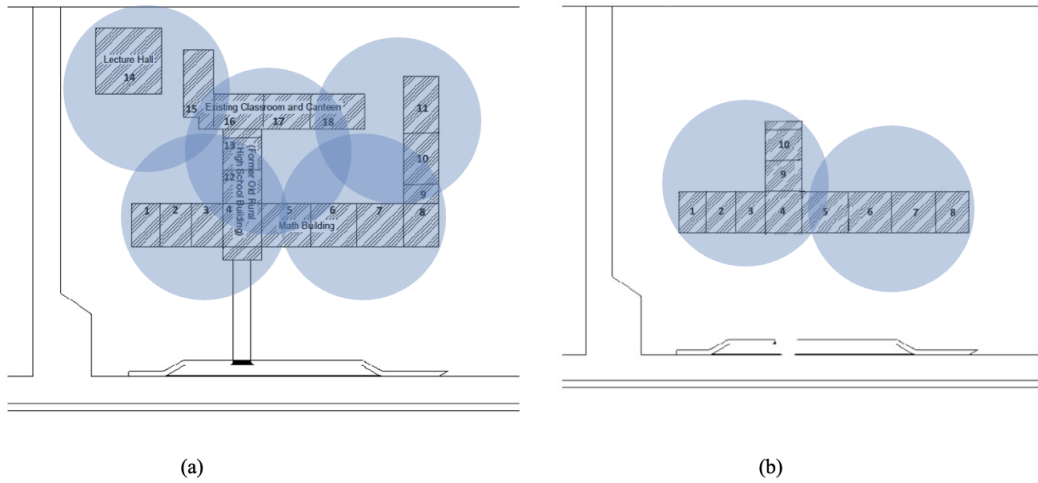


FIGURE 2. Old Math Building 1st (a) and 2nd (b) Floor Plan with Optimal Wi-Fi Placement

**Comparison to Dynamic Programming**

Using Linear Program Solver (LiPS), a software used on solving linear, integer and goal programming problems, results of ACO on SCP Program have been compared to that of LiPS to determine how their results vary.

For the 1sr floor of Math Building:

	Optimal Value	Run Time	Number of Iterations	Areas Covered
ACO on SCP Program	5	40 seconds*	160	100%
LiPS	5	.33 seconds	24	100%

TABLE 6. Comparison between results of ACO on SCP Program and LiPS on Math Building 1st Floor. \* The time it took to finish the whole number of iterations.

For the 2nd floor of Math Building:

	Optimal Value	Run Time	Number of Iterations	Areas Covered
ACO on SCP Program	2	11 seconds*	4	100%
LiPS	2	.07 seconds	13	100%

TABLE 7. Comparison between results of ACO on SCP Program and LiPS on Math Building 2nd Floor. \* The time it took to finish the whole number of iterations.

Both tables show that both programs obtained the same optimal value for each SCP although ACO on SCP Program may not be consistent since it only computes for a near-optimal value. It can also be observed that the running time for both programs show that they are quite on par with each other. Though ACOonSCP has a larger running time because it finishes 1000 iterations before producing an optimal solution, expanding the problem to a large number of variables (representative to the real-world problems) will give LiPS more computing time since it uses simplex to compute for the optimal solution. Not to mention, more difficulty in manually inputting values and the limitation on the number of variables and constraints that the LiPS program have (Hood, 2016). ACO on SCP Program is not consistent on number of iterations before producing an optimal result, but it is quite expected since it has a nature of a metaheuristic and operates based mainly on random numbers. Still, 100% of areas have been covered by both programs using the associated optimal solution they generated. Also, since the ants used in the study are “normal ants” in ACO terms, adding more ants/iterations to the ACOonSCP will not have that much effect on the result, and may even further prolong the iteration time (Johansson and Pettersson, 2018).

### Comparison to Current ITC Wi-Fi Set-up

According to the data from the University of the Philippines Los Baños Information Technology Center (UPLB- ITC), Math Building currently has 1 Wi-Fi router installed on a certain area which covers 20% of the whole Math Building. Meanwhile, based on the result of ACO on SCP Program, 7 are needed to cover the entire Math Building (1st Floor and 2nd Floor).

	Number of Routers	Areas Covered
ACO on SCP Program	7	100%
LiPS (Dynamic Programming)	1	20%

TABLE 8. Comparison between proposed number of ACO on SCP Program to current Wi-Fi Placement

## 5. SUMMARY AND CONCLUSION

In this research, Ant Colony Optimization Metaheuristic was applied to the Set Covering Problem. Single Row Oriented Method was used to generate solutions. Pheromone trails and heuristic information were updated to provide a faster and near-optimal solution. Solutions were returned, and the optimal solution was computed through a Scilab 5.4.1 program. Specifically, ACO was applied to a specific SCP. A program was implemented to solve the SCP. Then, data gathered from ITC and OVCPD were used to construct a SCP based on Optimal Allocation of Wi-Fi on the Math Building and was solved through

the program. Results have shown that to cover the entire MB, seven (7) 802.11n Wi-Fi Routers are needed. Computational results show that the proposed algorithm is efficient in generating high quality solutions. Comparisons with Dynamic Program revealed that it is competitive in solving the SCP. The work done in this paper also demonstrates the versatility of ACO and its high potential in solving hard combinatorial optimization problems. For extensions of the paper, branch-and-bound or branch-and-cut algorithms can be used to solve these types of problems. Having a comparison between our approach, other ACO approaches, and other metaheuristics approach can help determine which metaheuristics are suited for certain SCPs.

## REFERENCES

- [1] B. Birbil, and K. Bulbul, *The set covering problem revisited: An empirical study of the value of dual information*, Journal of Industrial and Management Optimization **11**(2015), 575-594.
- [2] C. Blum, and A. Roli, *Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison*, ACM Comput. Surv. **35**(2001), 268-308.
- [3] I. Brezina Jr., and Z. Cickova, *Solving the Travelling Salesman Problem Using the Ant Colony Optimization*, International Scientific Journal of Management Information Systems **6**(2011), 10-14.
- [4] D. Buezas, *Constraint-based modeling of Minimum Set Covering: Application to Species Differentiation*, Master Thesis, Faculdade de Ciencias e Tecnologia, Departamento de Informatica, Universidade Nova de Lisboa, 2010.
- [5] G. Cornuejols, M. Trick and M. Saltzman, A Tutorial on Integer Programming. <http://www.math.clemson.edu/~mjs/courses/mthsc.440/integer.pdf>, accessed January 18, 2022
- [6] M. Dorigo, and T. Stutzle, *Ant Colony Optimization*, Computational Intelligence Magazine, IEEE, **1**(2006), 28-39.
- [7] M. Dorigo, M. Birattari, and T. Stutzle, *Ant Colony Optimization: Artificial Ants as a Computational Intelligence Technique*, IEEE Computational Intelligence Magazine **1**(2006), 28-39.
- [8] M. Finkler, T. Stützle, and H. Lourenço, *Exploiting fitness distance correlation of set covering problems*, Lecture Notes In Computer Science, **2279**(2002), 61-71.
- [9] R. Impagliazzo, S. Lovett, R. Paturi, and S. Schneider, *0-1 Integer Linear Programming with a Linear Number of Constraints*, arXiv:1401.5512, (2014).
- [10] G. Lan, G. DePuy, and G. Whitehouse, *An effective and simple heuristic for the set covering problem*. European Journal of Operational Research **176**(2005), 1387-1403.
- [11] L. Lessing, T. Dumitrescu, and T. Stutzle, *A Comparison Between ACO Algorithms for the Set Covering Problem*, Springer-Verlag Berlin Heidelberg, 2004.
- [12] S. Olafsson, *Chaper 21 Metaheuristics*, Handbook on Simulation, Handbooks in Operations Research and Management Science, **13**(2006), 633-654.
- [13] Z. Ren, Z. Feng, L. Ke, and Z. Zhang, *New ideas for applying ant colony optimization to the set covering problem*, Computers & Industrial Engineering **58**(2008), 774-778.
- [14] N. Rokbani, A. Momasso, and A. Alimi, *Ant Supervised by PSO Meta-heuristic with Application to TSP*, Proceedings Engineering & Technology **4**(2013), 148-152.
- [15] K. Schiff, *Ant Colony Optimization Algorithm for the Set Covering Problem*. Technical Transactions Automatic Control, **1-AC** (2013).
- [16] M. Schulze, *Linear Programming for Optimization*, Perceptive Scientific Instrument, Inc., 1998.
- [17] T. Stutzle, and H. Hoos, *Max-min ant system*, Future Generation Computer Systems, **16**(2000), 889-914.
- [18] W. Tfaili, and P. Siarry, *A new charged ant colony algorithm for continuous dynamic optimization*, Applied Mathematics and Computation, **197**(2008), 604-613.
- [19] P. Varaiya, *Lecture Notes on Optimization*, Van Nostrand Reinhold Notes on System Sciences, 1998.
- [20] K. Vassilis, *Linear Programming*. <https://docplayer.net/27853871-Vassilis-kostoglou-url-linear-programming-1.html>, accessed January 18, 2022.
- [21] L. Pettersson and C. Lundell Johansson, *Ant Colony Optimization - Optimal Number of Ants*. <https://www.diva-portal.org/smash/get/diva2:1214402/FULLTEXT01.pdf>, accessed January 18, 2022.
- [22] S. Hood, *Linear Program Solver*, <https://silo.tips/download/linear-program-solver>, accessed January 18, 2022.
- [23] V. Selvi, and R. Umarani, *Comparative Analysis of Ant Colony and Particle Swarm Optimization Techniques*, International Journal of Computer Applications **5**(2010), 908-1286.
- [24] N. Abd-alsabour, H. Hefny and A. Moneim, *Heuristic information for ant colony optimization for the feature selection problem*, IEEE Conference Anthology, (2013), 1-5. doi:10.1109/ANTHOLOGY.2013.6784795.

CORRESPONDING AUTHOR, INSTITUTE OF MATHEMATICAL SCIENCES AND PHYSICS, UNIVERSITY OF THE PHILIPPINES LOS BANOS, LAGUNA, PHILIPPINES, 4030

DEPARTMENT OF MATHEMATICS, UNIVERSITY OF HOUSTON, TX, USA, 77204

*Email address:* [chbuhat@uh.edu](mailto:chbuhat@uh.edu)

INSTITUTE OF MATHEMATICAL SCIENCES AND PHYSICS, UNIVERSITY OF THE PHILIPPINES LOS BANOS, LAGUNA, PHILIPPINES, 4030

*Email address:* [jersonvillamin@gmail.com](mailto:jersonvillamin@gmail.com)

INSTITUTE OF MATHEMATICAL SCIENCES AND PHYSICS, UNIVERSITY OF THE PHILIPPINES LOS BANOS, LAGUNA, PHILIPPINES, 4030

*Email address:* [gacuaresma@up.edu.ph](mailto:gacuaresma@up.edu.ph)