Hybridising Ant Colony Optimisation with a Upper Confidence Bound algorithm for Routing and Wavelength Assignment in an Optical Burst Switching network

Andrew S. Gravett
Department of Computing Sciences
Nelson Mandela Metropolitan University
Port Elizabeth, South Africa
Email: andrew.gravett@nmmu.ac.za

Mathys C. du Plessis
Department of Computing Sciences
Nelson Mandela Metropolitan University
Port Elizabeth, South Africa
Email: mc.duplessis@nmmu.ac.za

Timothy B. Gibbon
Department of Physics
Nelson Mandela Metropolitan University
Port Elizabeth, South Africa
Email: tim.gibbon@nmmu.ac.za

***Abstract*—Ant Colony Optimisation (ACO) has been extensively applied to the network routing problem. Simulated ants are used to explore the network while recording information regarding their success by means of pheromones that are deposited on the route. A balance must be found between exploration of new routes and exploitation of established routes. Modern Monte Carlo game play algorithms, like Upper Confidence Bound applied to Trees (UCT), also have to decide which game branches to explore and which solutions should be exploited. The Upper Confidence Bound 1 (UCB1) formula is used to choose move branches, thus creating a balance between exploration and exploitation. This paper investigates the use of the UCB1 formula in an ACO algorithm to determine which routes should be selected. UCB1 was incorporated into an ACO algorithm that allocates a path (from source to destination) and an appropriate wavelength to packets to be routed in a network, which employs Optical Burst Switching (OBS). The new algorithm was evaluated against an existing ant-based algorithm on three network topologies in order to determine its effectiveness. Results obtained indicated that the proposed algorithm outperformed the existing algorithm in most scenarios.**

I. Introduction

ACO is a bio-inspired system based on the social and foraging behaviour patterns of ants seeking an optimal path between their food source and colony [1]. This behaviour has been imitated and adapted for use in solving various graph based computational problems. ACO algorithms have been extensively applied to solve routing problems within telecommunication networks [2]. ACO is of particular interest for routing problems as it is run continuously adapting to changes in the network state and traffic load in real time. Several ACO algorithms have been used for solving the routing and wavelengths assignment problem in Wavelength Division Multiplexing (WDM) optical networks [3]–[5].

Monte Carlo tree search is often applied to game playing algorithms. The branches of the tree consist of all possible moves that can be made by the players. These trees are too large to be exhaustively searched (for all but the simplest games). Monte Carlo based algorithms randomly explore branches and then make a move based on the ratio of the random games that yielded a victory for the player. One of the most effective and widely applied variant of these algorithms is the UCT algorithm [6]. The unique aspect of UCT is that branches are not explored purely randomly. Rather, promising branches that are more likely to result in victory are explored in more detail. The mechanism that is used to guide the search process is the UCB1 formula of Auer et al. [7]. The UCB1 formula was initially developed to solve the N-Armed Bandit problem [8], which refers to a gambler being faced with a series of slot machines, and having to determine which machines to play more often so as to maximise winnings. The UCB1 policy balances the exploration of the game tree with the exploitation of promising branches in the UCT algorithm. This paper investigates the hybridisation of ACO with UCB1. The UCB1 policy is used to determine paths and wavelengths that ant following in the network. The formula ensures that all paths are explored, but that more successful paths are selected more often.

OBS is a technology that offers a more flexible and dynamic optical network compared to traditional Optical Circuit-Switched (OCS) networks [9], [10]. However, OBS does suffer from burst losses due to a lack of cost effective optical buffers and resource reservation schemes. The Routing and Wavelength Assignment (RWA) is especially difficult in an optical network with no wavelength converters and buffers within the core increasing the risk of burst losses. In an optical network with the Wavelength Continuity (WC) constraint, the RWA process must be efficient to minimise burst losses and make effective use of network resources.

This work explores the use of an entire path, route and wavelength combination, from source to destination as a method for RWA. Contributing a distributed RWA protocol which balances exploitation and exploration within a bufferless dynamic OBS network under the wavelength continuity constraint (same wavelength must be used during traversal within the network from source node to destination node).

This paper is organised as follows: Section II outlines the background of OBS, RWA and introduces the motivation behind this work. Section III details an existing ant-based protocol. Section IV describes in detail the proposed RWA protocol and its implementation. Section V explains the set-up of the simulation and analysis of the gathered results. Section VI presents a summary of main conclusions for the paper.

II. Background and Motivation

OBS is an optical switching paradigm designed to achieve increased bandwidth flexibility with a lack of efficient optical buffers. In OBS, instead of converting and transmitting a single packet of data, the packets are aggregated into large chunks of data at the edge of the network and transmitted in the form of a burst [10]. The bursts are preceded by a Burst Control Packet (BCP) which is responsible for reserving the optical resources [11]. The bursts of data are transmitted using a wavelength channel and are switched through the core nodes of the network. The bursts are later disassembled at the destination node and its packets are delivered accordingly [12]. Selection of appropriate optical resources is the RWA problem [13]. With respect to the aforementioned, routes through the network are calculated according to some form of heuristic applied to the network. However, each of the links within the network have multiple wavelengths each of which are capable of carrying different bursts. To satisfy the burst request, it is necessary to assign a route and wavelength while avoiding wasting resources and utilising good network management. Many algorithms have been applied to RWA in OBS [3]–[5]. In particular, ant-based algorithms have motivated many studies on this topic. This is because ACO is able to run continuously and adapt to changes in real-time which provides a dynamic solution to the RWA problem. Furthermore, the use of ants traversing the network is analogous of OBS BCPs reserving optical resources therein.

In previous works [3]–[5], the route and wavelength selection are treated separately, such that the route is often calculated on a hop for hop basis with initially either first-fit or random wavelength assignment. To our knowledge, pheromones for an entire path have not been investigated, a route and wavelength tuple in OBS. We propose investigating the use of a tuple for the RWA problem. The tuple would contain a wavelength, route, and the number of success and failures. Each tuple in a set of tuples would have its own reward distribution independent of all the other tuples. This can be considered to be a bandit-problem. The algorithm would essentially need to select one of K possible tuples that would maximise the reward or likelihood of successful transmission through the network. The choice of the tuple can thus be seen as a separate decision problem. The reward of each tuple is not known initially. This exploitation-exploration scenario is a trade off between searching tuples discovered to have a higher reward and tuples that exhibit lower reward but could have been classified incorrectly.

In a bandit-problem, if the upper confidence bound (UCB) of optimality of each arm is known, regret can be minimised.

TABLE I: Pheromone Table [3]

		Output Link	
Input Link	**Wave**	Link1	Link 2
Link 1	1	$\tau_{1,1,1}$	$\tau_{1,2,1}$
	2	$\tau_{1,1,2}$	$\tau_{1,2,2}$
	3	$\tau_{1,1,3}$	$\tau_{1,2,3}$
	4	$\tau_{1,1,4}$	$\tau_{1,2,4}$
Link 2	1	$\tau_{2,1,1}$	$\tau_{2,2,1}$
	2	$\tau_{2,1,2}$	$\tau_{2,2,2}$
	3	$\tau_{2,1,3}$	$\tau_{2,2,3}$
	4	$\tau_{2,1,4}$	$\tau_{2,2,4}$

The upper confidence bound is a statistical measure of the highest confidence interval of an unknown distribution. Auer et al. [7] proposed a policy, named UCB1, that minimises this regret by assigning a penalty to moves that have been played too often, thus providing an automatic means of selecting when to exploit good moves, and when exploration becomes favourable. UCB1 allows the regret to grow logarithmically over the span of c_T plays without a priori knowledge of the distribution of each arm. The arm with the highest UCB is played calculated using the following equation:

$$UCB(j) = \bar{\mu_j} + C\sqrt{\frac{2\ln(c_T)}{c_j}} \tag{1}$$

where $\bar{\mu_j}$ is the mean reward of arm j, c_j is the total number of times arm j has been played, and c_T indicates the total number of plays so far. The two terms of Equation 1 encourage exploitation and exploration respectively. The mean reward of the arm $\bar{\mu_j}$ encourages UCB1 to play moves that have been demonstrated to be effective. $\sqrt{\frac{2\ln(c_T)}{c_j}}$ is a penalty term to encourage exploration of less visited branches by weighting branch cumulative payout values by the number of times that they have been played. An additional scaling C term is added in practice to the decaying penalty term to further control the exploration and exploration rates of the approach.

III. ACRWA

Ant Colony Routing and Wavelength Assignment (ACRWA) utilises a distributed approach optical resource assignment in which the ant colonies shared information with one another [3]. ACRWA describes an ant colony based solution to the RWA problem for dynamic WDM OBS networks with the WC constraint. ACRWA performs an initialisation process at each node in the network. This process populates a list of neighbour nodes, the available wavelengths and generates a list of k-shortest routes to all the remaining nodes within the network with each route using a different output port. The pheromone table for each node in the network consists of pheromone values assigned to every switching configuration of input port i, output port j and wavelength k at a node. Table I is an example of an initialised pheromone table for a node with 2 ports, in a network with 4 wavelengths.

The RWA protocol governing the specific behaviour of the ants begins once the initialisation process has been completed. This protocol is run by the nodes within the network for each new transmission request. The objective of the algorithm is to select the most beneficial output port u and wavelength λ combination to assign the ant. This assignment is a greedy selection of the u and λ pair that has the greatest benefit calculated using the product of desirability of a certain output port and pheromone deposition described by the following equation:

$$u, \lambda = \{argmax\{\tau_{ijk}(t)\eta_{nj}^{\beta}(t)\} | j \in S_n^m(t), k \in D_n^m(t)\}$$

where u is the selected output port and λ the selected wavelength. τ_{ijk} is the pheromone level stored at the node for input port i and output port j on wavelength k and where D_n^m is the set of existing wavelengths at node n that have been initialized for destination m. S_n^m is the set of currently available and feasible nodes from n to m. η_{nj} is equal to the reciprocal of a function of the shortest path from the the source to destination node through output port j. β is a constant used to emphasize the use of shorter paths. After the initial selection, the transition rule is responsible for selecting the next hop (output link) in the ant routing process toward its destination. The user-specified parameter $r_0 \in [0, 1]$ is used to balance between exploitation and exploration. If it is determined the ant must exploit, the algorithm selects the output port with the greatest value using the following equation:

$$u = argmax_{j \in S_n^m(t)}\{\tau_{ijk}(t)\eta_{nj}^{\beta}(t)\}$$

Otherwise, the ant will choose to explore by iterating through each output port j assigning it a probability as follows:

$$P_{iuk}(t) = \frac{\tau_{iuk}(t)\eta_{nu}^{\beta}(t)}{\sum_{j \in S_n^m} \tau_{ijk}(t)\eta_{nj}^{\beta}(t)}$$

The output port is selected based on an empirical distribution based on the assigned probabilities above. ACRWA ants are implemented as the BCP. The BCPs perform the RWA and update the pheromone concentrations while reserving optical resources for the burst. Successful reservation performs a local update using equation:

$$\tau_{ijk}(t+1) = \tau_{ijk}(t) + \alpha e^{-\phi \Delta l}$$

where α and ϕ are user-specified values. The feedback BCP backtracks the path of the ant and updates the pheromone trails using the global update as follows:

$$\tau_{ijk}(t+1) = (1-\rho)\tau_{ijk}(t) + \gamma_{ij}\rho\Delta\tau_{ijk}$$

where ρ is a user-specified value for the pheromone evaporation coefficient. $\gamma_{ij} = 1$ for a successful reservation otherwise $\gamma_{ij} = -1$. The amount of deposited pheromone $\Delta\tau_{ijk} = e^{-\omega \Delta l}$ depends on the difference in length Δl of the reverse path followed by the feedback ant so far and the length of the shortest path to the origin of that ant from the current node that processes it where ω is a user-specified value.

TABLE II: Routing Table

Route	Wave	Success	Fail
AB	1	59	0
ACB	2	2	33
AB	3	22	19
AC	1	16	4
ABC	2	103	7
AC	3	42	25

IV. UCBRWA

The proposed Upper Confidence Bound Routing and Wavelength Assignment (UCBRWA) algorithm describes a distributed protocol which treats the combination of a wavelength and a route to destination as a single candidate solution (or tuple). The different network nodes do not share information and function independently of one another.

A. Routing Table

Each node, n, in the network has a routing table RT_n. The updating and assignment rules make use of the routing table. The routing table, RT, contains a number of candidate solutions ε_i. Each candidate solution, ε_i, is a tuple $< \mathcal{N}_{nmi}, \lambda_i, \kappa_i, \chi_i >$. A tuple is composed of a route $\mathcal{N}_{nmi}$ from node n to m, wavelength λ_i, number of successes κ_i and number of failures χ_i. Table II gives an example of the routing table for a node, A, with 2 links in a network running 3 wavelength channels.

B. Initialisation

The UCBRWA protocol is composed of two main algorithms, the initialization and routing algorithms. The initialization algorithm, Algorithm 1, is responsible for running the first stage of the protocol. This includes the generation of routing information, configuring the UCBRWA parameters, setting up the routing tables and managing traffic generation in the network. N represents the set of all network nodes. Local routing information is generated for each node $n \in N$ in the network. The routing table is formed by calculating for every possible destination $m \in (N \setminus \{n\})$, a candidate list $\mathcal{N}_{nm}$ of k-shortest routes at node n. A routing table RT_n for each node $n \in N$ is initialised. This process is performed by creating candidate solutions and adding them to the routing table. The routing table RT_n, for node n is composed of J randomly generated candidate solutions. A total of $\frac{J}{|N|-1}$ candidate solutions are generated for each destination node $m \in (N \setminus \{n\})$. Each candidate solution is composed of a destination node m, a randomly selected route from the candidate list $\mathcal{N}_{nm}$ from node n to m and the wavelength is generated from a random number within the valid wavelength range. Once all of these steps have been completed, the network nodes are ready to process burst transmissions.

C. Routing and Wavelength Assignment

The RWA algorithm, Algorithm 2, is run by each of the nodes within the network for each new data transmission request. The objective of the algorithm is to calculate the best candidate solution for the transmission request. The candidate

Algorithm 1: Initialization

Variables: N
Initialize parameters: α_1, C, J
foreach *node* $n \in N$ **do**
  Initialize local routing information
  $m \in (N \setminus \{n\})$
  foreach *possible destination* m **do**
    Initialize candidate route list $\mathcal{N}_{nm}$
  end
  Initialize routing table RT_n
end
while *Burst to transmit* **do**
  Run RWA Algorithm
end

solution is selected by making use of the routing table at the source node. It is responsible for selecting a candidate solution, route and wavelength combination, for the burst routing process to the destination node.

Algorithm 2: Routing and Wavelength Assignment

$r \leftarrow random()$;
if $r < \alpha_1$ **then**
  Find $\varepsilon_i \in RT_{nm}$ using (2)
  Return λ_i and $\mathcal{N}_{nmi}$
else
  $\lambda \leftarrow$ select a random wavelength
  $\mathcal{N} \leftarrow$ select a random route from $\mathcal{N}_{nm}$
  Return λ and $\mathcal{N}$

For exploitation and exploration, the objective is to calculate the best possible route wavelength combination ε. The selection process makes use of the RT_{nm} using Equation (2). RT_{nm} is the set of valid route wavelength combinations to forward a burst, avoiding loops or ports without a feasible route from source n to destination m.

$$\hat{\varepsilon} = argmax_{\varepsilon_j \in RT_{nm}} \frac{\kappa_j}{\kappa_j + \chi_j} + C\sqrt{\frac{2\ln(\kappa_{nm} + \chi_{nm})}{\kappa_j}} \quad (2)$$

where C is a user-specified value for exploration and $\frac{\kappa_j}{\kappa_j+\chi_j}$ is the mean reward for ε_j. κ_j is the number of successes for ε_j. χ_j is the number of failures for ε_j. $\kappa_{nm} = \sum_{\varepsilon_j \in RT_{nm}} \kappa_j$ is the total number of successes for all $\varepsilon_j \in RT_{nm}$. $\chi_{nm} = \sum_{\varepsilon_j \in RT_{nm}} \chi_j$ is the total number of failures for all $\varepsilon_j \in RT_{nm}$.

If creation is selected, it is determined that the algorithm must calculate a new candidate solution. During the creation of a new candidate solution, a random wavelength is selected and a route from source node n to destination node m is selected from $\mathcal{N}_{nm}$. This new candidate solution is given to the burst and added to the set of candidate solutions located in RT_{nm}. However, there is a set limit to the number of candidate solutions contained within the RT. The candidate solution ε_j with the lowest mean reward, $\frac{\kappa_j}{\kappa_j+\chi_j}$, where $\varepsilon_j \in RT_{nm}$ is removed and replaced with the newly created one. The limit ensures that low performing candidate solutions do not linger on and decrease network performance. Limiting the removal of a low performing candidate solutions from the same destination set as the newly created one, ensures that all destinations are given an equal allocation of candidate solutions. Otherwise, if the lowest overall candidate solutions were removed it would result in a poor performing destination not having a fair share of candidate solutions and vice versa.

D. OBS Implementation

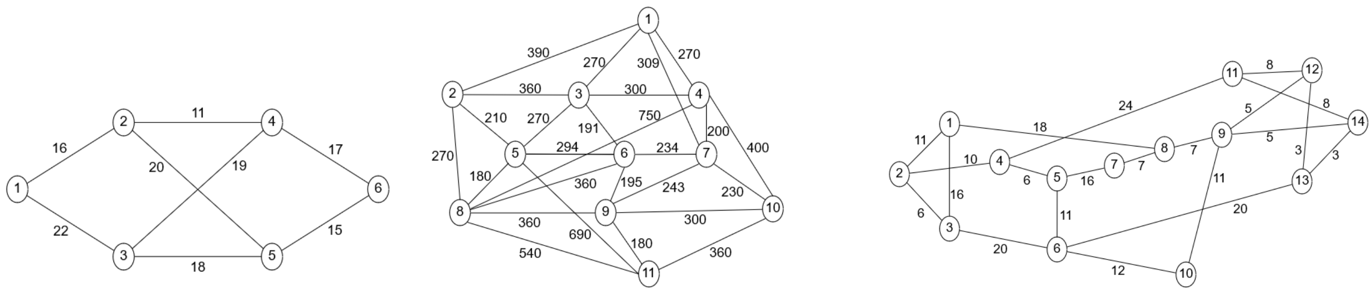
The BCP for each burst is responsible for reserving the optical resources. The OBS implementation makes use of a one way reservation scheme according to which the source node transmits the set-up request, the BCP, and then transmits the data burst without waiting for the reservation set-up acknowledgement. As soon as the BCP is received by the switch, the switching elements are configured for the coming burst. Until the intermediate node receives a BCP release acknowledgement (BCP-RA), the switching elements will remain in their current configuration [14]. Upon successful transmission of a data burst, the BCP, the destination node transmits the BCP-RA (a positive feedback) to the source of the burst. The BCP-RA releases the resource reservations and informs the burst source of successful transmission. This prevents further channel collisions, but places additional overhead to the network. A blocked BCP will initiate a BCP-RA (a negative feedback) on the reverse path initially followed by the BCP. The algorithm terminates once the BCP-RA arrives at the source of the burst transmission.

E. Global Update

The global update is performed by the BCP-RA. The BCP-RA contains the the information regarding the success or failure of the bursts delivery to the destination. The BCP-RA follows the reverse path of the BCP and only applies the update at the source of the BCP. The BCP-RA updates the χ and κ values for the candidate solutions within RT. κ_i is the count of the success of candidate solution ε_i and likewise χ_i is a count of the failure of the candidate solution. The values for χ_i and κ_i for candidate solution ε_i are increased in increments of 1.

V. Numerical Evaluation

This section presents the numerical results obtained in simulations to measure the performance of the routing protocol implementations in an OBS environment. Three different network topologies are adopted in the simulations. A small test network topology composed of 6 nodes with 8 links in Figure 1a, a medium test network topology of 11 nodes with 26 links in Figure 1b and a large test network topology of 14 nodes with 21 links in Figure 1c. The purpose of the different network topologies is to evaluate the performance of the protocol in a simple network, a more connected example (COST239 topology in Figure 1b) and also within a more realistic scenario (NSFNET-14 topology in Figure 1c).



(a) Small 6-Node network topology (b) Medium 11-Node network topology (c) Large 14-Node network topology

Fig. 1: Test Network Topologies

Each of the simulations were tested under various network loads. These simulations were performed a total of 30 times and the mean results calculated. Each simulation attempts to transmit $5X10^5$ bursts, including the initial learning phase. The source and destination nodes are randomly selected for each burst. Two values are measured throughout the duration of a simulation. The number of successful burst transmissions κ_T and the number of burst blocks χ_T. The success ratio R used as a performance measure is calculated as follows:

$$R = \frac{\kappa_T}{\chi_T + \kappa_T}$$

The routing algorithm used for comparison with UCBRWA was an existing ant based routing algorithm, ACRWA. The simulation test scenarios were performed without buffer and wavelength conversion capability. The bursts must fulfil the WC constraint and ensure that no burst is stored within a buffer for any length of time. The test scenario is a representation of a traditional WDM OBS network with a static grid in which no linear and non-linear effects are taken into account. The test simulations gather information on the number of successful and unsuccessful (or blocked) burst transmissions. The higher the number of successful burst transmissions that occur, the higher the measure of an algorithms performance. The parameters of the UCBRWA protocol used throughout the simulations are as follows: α_1 = 0.995, C = 2.0. J = 200 for the small, J = 400 for the medium and J = 700 for the large network topology. The parameters of the ACRWA protocol used throughout the simulations are a follows: ρ = 0.01, ω = 0.75, α = 0.001, ϕ = 0.75, β = 2.0. r_0 = 0.8 for the small, r_0 = 0.7 for the medium and r_0 = 0.8 for the large network topology.

A. Small Network

The first scenario considers the small network topology Figure 1a which has 8 available wavelengths on each of the links within the network configuration. Figure 2 shows a comparison of the success ratio performance of UCBRWA and ACRWA as a function of load on the network using 8 wavelengths. A clear differentiation can be observed regarding the performance of each of the protocols except between the load of 15 to 25. The performance of each of the protocols converge from a load of 15 and diverge once more at a load of 25. The limited number of wavelengths in this simulation under the higher network loads causes a lower burst success ratio. The congestion generated by the increasing load on the network is the main cause of this difference.

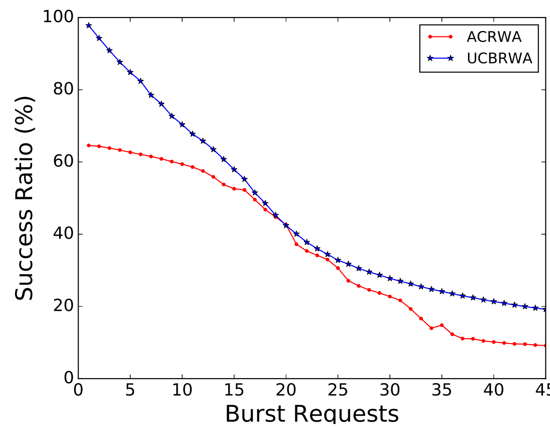


Fig. 2: Small Network Topology with 8 Wavelengths

TABLE III: Success ratio and 95% confidence interval at a specific load on the small network topology with 8 wavelengths.

Load	ACRWA	UCBRWA	p-value
5	62.67 ± 0.08	*84.83 ± 0.40*	0.000
10	59.40 ± 0.06	*70.35 ± 0.39*	0.000
15	52.58 ± 0.66	*57.91 ± 0.31*	0.000
20	*42.63 ± 0.69*	42.41 ± 0.27	0.003
25	30.61 ± 0.67	*32.80 ± 0.14*	0.000
30	22.74 ± 0.43	*27.75 ± 0.09*	0.000
35	14.81 ± 1.41	*24.15 ± 0.08*	0.000
40	10.14 ± 0.16	*21.35 ± 0.05*	0.000
45	9.16 ± 0.13	*19.18 ± 0.05*	0.000

The mean success ratios over 30 repeats of each algorithm are reported along with the 95% confidence interval in Table III. Outcomes of Mann-Whitney U tests comparing the results of ACRWA and UCBRWA are included. Results which are statistically significantly better are printed in italics. At a load of 10, UCBRWA is approximately 10.95% higher than ACRWA. At a load of 20, ACRWA is approximately 0.22% higher than UCBRWA. At a load of 30, UCBRWA is approximately 5.01% higher than ACRWA. At a load of

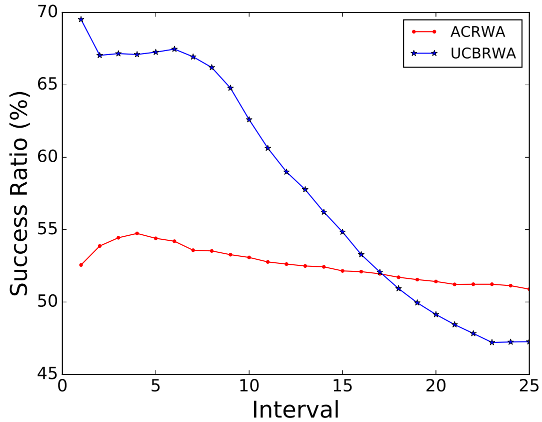


Fig. 3: Success ratio over time in the small network topology at a load of 15

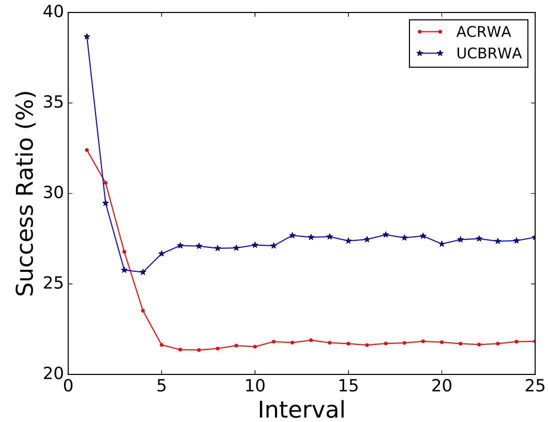


Fig. 4: Success ratio over time in the small network topology at a load of 30

40, UCBRWA is approximately 11.21% higher than ACRWA. On average, UCBRWA performs approximately 9.84% higher than ACRWA.

Figure 3 is an illustration of the algorithms adapting to a network load of 15 over the duration of the simulation. UCBRWA is initially highly successful, thereafter it steadily declines in performance and stabilizes below that of ACRWA. ACRWA has an small initial increase which stabilizes into a very gradual decline.

Figure 4 is an illustration of the algorithms adapting to a network load of 30 over the duration of the simulation. Under this higher load on the network, UCBRWA plummets initially but stabilizes into a gradual increase. ACRWA has an almost identical trend with lower performance, stabilizing with no noticeable increase.

From Figure 3, it can be seen that UCBRWA has an initial high performance which decreases to below that of ACRWA. At the lower loads (up to a load of approximately 25) UCBRWA performs better than ACRWA, but the values are skewed due to UCBRWAs initial success during the simulation. Once the high loads (above a load of approximately 25) are placed on the network, UCBRWA as seen in Figure 4 was able to perform better. ACRWA and UCBRWA do have similar performances from a load of 15 to 25. Overall, UCBRWA is more successful in the small network topology.

B. Medium Network

The second scenario considers the medium network topology Figure 1b which has 12 available wavelengths on each of the links within the network configuration. Figure 5 shows a comparison of the success ratio performance of UCBRWA and ACRWA as a function of load on the network using 12 wavelengths. Clearly, the results gathered on the medium network topology are higher than that of the small network topology. The medium network differs from the others topologies as it is more connected and makes use of more wavelengths per link creating a larger solution space.

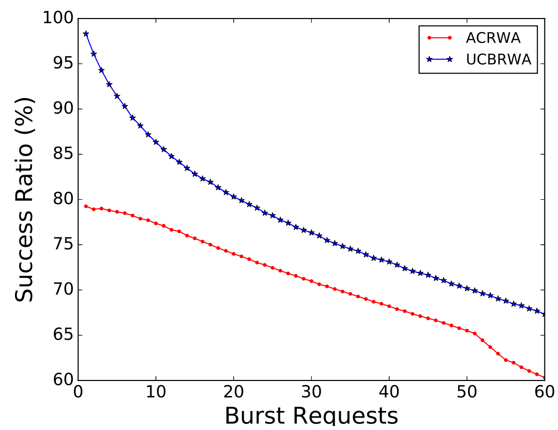


Fig. 5: Medium Network Topology with 12 Wavelengths

TABLE IV: Success ratio and 95% confidence interval at a specific load on the medium network topology with 12 wavelengths.

Load	**ACRWA**	**UCBRWA**	**p-value**
5	78.64 ± 0.12	*91.42 ± 0.11*	0.000
10	77.36 ± 0.07	*86.34 ± 0.14*	0.000
15	75.71 ± 0.06	*82.82 ± 0.12*	0.000
20	73.99 ± 0.06	*80.30 ± 0.11*	0.000
25	72.45 ± 0.06	*78.23 ± 0.12*	0.000
30	70.98 ± 0.05	*76.33 ± 0.11*	0.000
35	69.57 ± 0.05	*74.54 ± 0.10*	0.000
40	68.20 ± 0.05	*73.11 ± 0.09*	0.000
45	66.87 ± 0.04	*71.65 ± 0.08*	0.000
50	65.51 ± 0.06	*70.17 ± 0.08*	0.000
55	62.26 ± 0.17	*68.79 ± 0.09*	0.000
60	60.32 ± 0.07	*67.32 ± 0.08*	0.000

The mean success ratios over 30 repeats of each algorithm are reported along with the 95% confidence interval in Table IV. Outcomes of Mann-Whitney U tests comparing the results of ACRWA and UCBRWA are included. Results which are statistically significantly better are printed in italics. At a load of 15 UCBRWA is approximately 7.11% higher than ACRWA.

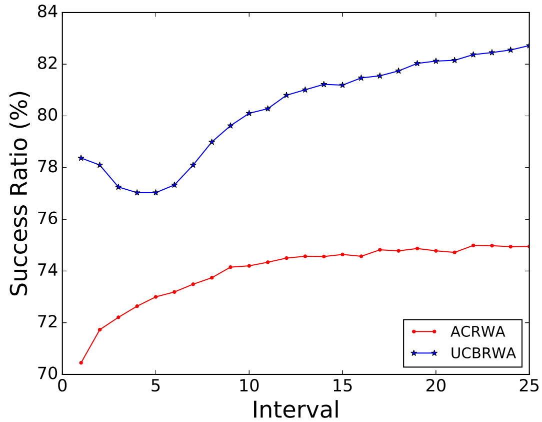


Fig. 6: Success ratio over time in the medium network topology at a load of 20

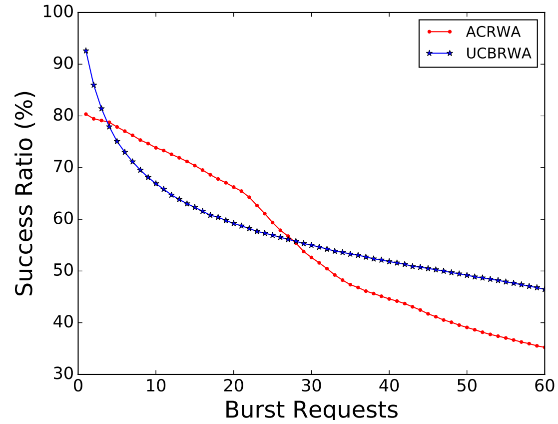


Fig. 7: Large Network Topology with 16 Wavelengths

At a load of 30 UCBRWA is approximately 5.35% higher than ACRWA. At a load of 45 UCBRWA is approximately 4.78% higher than ACRWA. At a load of 60 UCBRWA is approximately 7.0% higher than ACRWA. On average, UCBRWA performs approximately 6.97% higher than ACRWA.

Figure 6 is an illustration of the algorithms adapting to a network load of 20 over the duration of the simulation. ACRWA steadily increases in performance over the duration of the simulation, stabilizing at the end. UCBRWA has an initial decrease which levels out, leading to a steady increase in performance to the end. Similar trends were present at the other load levels on the medium network topology.

Noticeably, UCBRWA is more successful in the medium network topology. ACRWA is unable to perform as well as UCBRWA.

C. Large Network

The third scenario considers the large network topology Figure 1c which has 16 wavelengths on each of the links within the network. Figure 7 displays the behaviour of the UCBRWA and ACRWA algorithms in terms of success ratio as a function of network load. This is a comparison of the algorithms on a network similar to the NSFNET-14 node network. Notably, the success ratios in the large network topology are lower than that of the medium network topology. The large network has more nodes and wavelengths per link, but is less connected.

The mean success ratios over 30 repeats of each algorithm are reported along with the 95% confidence interval in Table V. Outcomes of Mann-Whitney U tests comparing the results of ACRWA and UCBRWA are included. Results which are statistically significantly better are printed in italics. At a load of 15 ACRWA is approximately 8.07% higher than UCBRWA. At a load of 30 UCBRWA is approximately 2.38% higher than ACRWA. At a load of 45 UCBRWA is approximately 8.77% higher than ACRWA. At a load of 60 UCBRWA is approximately 11.22% higher than ACRWA.

TABLE V: Success ratio and 95% confidence interval at a specific load on the large network topology with 16 wavelengths.

Load	ACRWA	UCBRWA	p-value
5	*77.86 ± 0.15*	75.07 ± 0.11	0.000
10	*73.83 ± 0.15*	66.90 ± 0.11	0.000
15	*70.39 ± 0.09*	62.32 ± 0.08	0.000
20	*66.23 ± 0.09*	59.20 ± 0.07	0.000
25	*59.37 ± 0.29*	56.93 ± 0.08	0.000
30	52.62 ± 0.21	*55.00 ± 0.07*	0.000
35	47.36 ± 0.13	*53.29 ± 0.05*	0.000
40	44.61 ± 0.05	*51.82 ± 0.05*	0.000
45	41.72 ± 0.22	*50.49 ± 0.05*	0.000
50	39.09 ± 0.09	*49.19 ± 0.03*	0.000
55	37.05 ± 0.07	*47.91 ± 0.03*	0.000
60	35.24 ± 0.05	*46.46 ± 0.05*	0.000

Figure 8 is an illustration of the algorithms adapting to a network load of 20 over the duration of the simulation. ACRWA has a rapid increase in performance stabilizing approximately halfway. UCBRWA has an initial decrease which levels out, stabilizing into a steady increase in performance to the end.

Figure 9 is an illustration of the algorithms adapting to a network load of 40 over the duration of the simulation. UCBRWA initially decreases in performance which stabilizes into a gradual increase in performance to the end. ACRWA has a rapid increase in performance that levels out and stabilizes for the duration of the simulation.

For the lower loads placed on the network, ACRWA outperforms UCBRWA. In Figure 8, ACRWA increases in performance from the beginning. UCBRWA however, only increases in performance towards the end, stopping on an upwards trend as UCBRWA took more time to learn as compared to ACRWA. UCBRWA performs much better under the higher loads which can also be seen in Figure 9.

The algorithms performances cross each other on two occasions, the start and mid way through the loads placed on the network. UCBRWA has a better initial performance which is short lived, being overtaken by ACRWA at a load of 4 placed on the network. ACRWA remains ahead in performance up to a load of 27 placed on the network. Thereafter, UCBRWA has an improved performance with an increase in load placed on

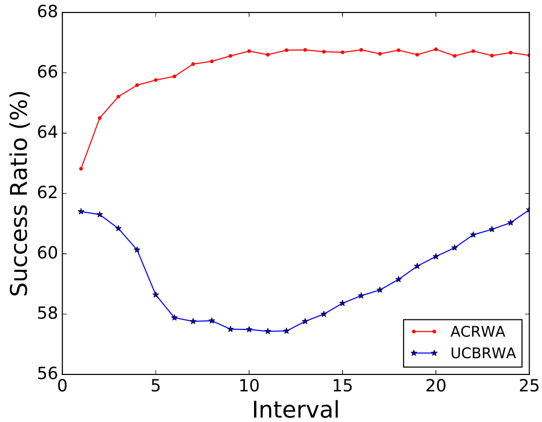


Fig. 8: Success ratio over time in the large network topology at a load of 20

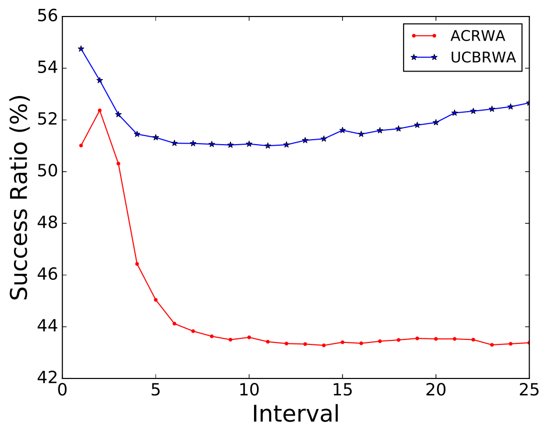


Fig. 9: Success ratio over time in the large network topology at a load of 40

the network compared to that of ACRWA.

A trend can be seen with respect to Figures 6, 8, 9 and to some respect in 4. These above mentioned figures are an illustration of UCBRWA ability to learn over time and make use of good candidate solutions. UCBRWA initially explores the various candidate solutions within the routing table, thereafter exploiting the more successful candidate solutions. Therefore, UCBRWA effectively explores and establishes a good set of candidate solutions.

VI. Conclusion

This work proposed an algorithm, UCBRWA, which investigated using an entire path (wavelength and route tuple) from source to destination. UCB was used to select the path providing a balance between exploitation and exploration of the tuples similar to that of the bandit problem. The protocol was evaluated on three separate network topologies in a bufferless OBS network with a WC constraint. In the small network scenario UCBRWA performed better than ACRWA under low and high loads, performing similarly under medium load. UCBRWA performed significantly better than ACRWA for all the loads tested on the medium network scenario. In the large network scenario ACRWA was able to perform better than UCBRWA under lower loads, however, UCBRWA faired better than ACRWA under high loads. In the evaluation, UCBRWA displayed the tendency of initially exploring candidate solutions, thereafter exploiting the successful solutions which resulted in UCBRWA having a better performance under high loads on the tested network scenarios. When compared to an existing ant-based algorithm (ACRWA) the proposed protocol displayed similar and improved performance throughout most of the test network scenarios.

Acknowledgment

The financial assistance of the National Research Foundation (NRF) and CISCO towards this research is hereby acknowledged. Opinions expressed and conclusions arrived at, are those of the authors and are not necessarily to be attributed to the NRF or CISCO.

References

[1] M. Dorigo and G. Di Caro, "Ant colony optimization: a new meta-heuristic," pp. 1470–1477, 1999. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=782657

[2] S. H. Ngo, X. Jiang, and S. Horiguchi, "An ant-based approach for dynamic rwa in optical wdm networks," *Photonic Network Communications*, vol. 11, no. 1, pp. 39–48, 2006. [Online]. Available: http://dx.doi.org/10.1007/s11107-006-5322-2

[3] J. Triay and C. Cervello-Pastor, "An ant-based algorithm for distributed rwa in optical burst switching," in *2009 11th International Conference on Transparent Optical Networks*, June 2009, pp. 1–4.

[4] J. Pedro, J. Pires, J. P. Carvalho, N. Siemens, and R. I. Siemens, "Ant Colony Optimization for Distributed Routing Path Optimization in Optical Burst-Switched Networks," *Ifsa-Eusflat 2009*, pp. 178–183, 2009.

[5] E. A. Donato, J. C. Júnior, and S. S. Vieira, "A Proposal of Dynamic RWA Using Ant Colony in Optical Burst Switched Networks," no. c, 2012, pp. 246–252.

[6] L. Kocsis and C. Szepesvri, "Bandit based monte-carlo planning," in *In: ECML-06. Number 4212 in LNCS*. Springer, 2006, pp. 282–293.

[7] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine Learning*, vol. 47, no. 2, pp. 235–256, 2002. [Online]. Available: http://dx.doi.org/10.1023/A:1013689704352

[8] A. F. V. Michael N. Katehakis, "The multi-armed bandit problem: Decomposition and computation," *Mathematics of Operations Research*, vol. 12, no. 2, pp. 262–268, 1987. [Online]. Available: http://www.jstor.org/stable/3689689

[9] S. Yao, B. Mukherjee, and S. Dixit, "Advances in photonic packet switching: an overview," *IEEE Communications Magazine*, vol. 38, no. 2, pp. 84–94, Feb 2000.

[10] Y. Chen, C. Qiao, and X. Yu, "Optical burst switching: a new area in optical networking research," *IEEE Network*, vol. 18, no. 3, pp. 16–23, May 2004.

[11] M. A. Aydin, T. Atmaca, H. Zaim, O. C. Turna, and V. H. Nguyen, "Performance study of obs reservation protocols," in *Telecommunications, 2008. AICT '08. Fourth Advanced International Conference on*, June 2008, pp. 428–433.

[12] J. J. P. C. Rodrigues and B. Vaidya, "Evaluation of resource reservation protocols for ip over obs networks," in *2009 11th International Conference on Transparent Optical Networks*, June 2009, pp. 1–4.

[13] K. Bhaskaran, J. Triay, and V. M. Vokkarane, "Dynamic anycast routing and wavelength assignment in wdm networks using ant colony optimization (aco)," in *2011 IEEE International Conference on Communications (ICC)*, June 2011, pp. 1–6.

[14] J. Y. Wei and R. I. McFarland, "Just-in-time signaling for wdm optical burst switching networks," *Journal of Lightwave Technology*, vol. 18, no. 12, pp. 2019–2037, Dec 2000.