Preprint. Accepted for publication in Evolutionary Computation Journal, 2015.

Hybrid Evolutionary Approaches to Maximum Lifetime Routing and Energy Efficiency in Sensor Mesh Networks

Alma A. M. Rahat A.A.M.Rahat@exeter.ac.uk
Department of Computer Science, University of Exeter, UK

Richard M. Everson R.M.Everson@exeter.ac.uk
Department of Computer Science, University of Exeter, UK

Jonathan E. Fieldsend J.E.Fieldsend@exeter.ac.uk
Department of Computer Science, University of Exeter, UK

Abstract

Mesh network topologies are becoming increasingly popular in battery powered wireless sensor networks, primarily due to the extension of network range. However, multi-hop mesh networks suffer from higher energy costs, and the routing strategy employed directly affects the lifetime of nodes with limited energy resources. Hence when planning routes there are trade-offs to be considered between individual and system-wide battery lifetimes. We present a multi-objective routing optimisation approach using hybrid evolutionary algorithms to approximate the optimal trade-off between minimum lifetime and the average lifetime of nodes in the network. In order to accomplish this combinatorial optimisation rapidly, our approach prunes the search space using k-shortest path pruning and a graph reduction method which finds candidate routes promoting long minimum lifetimes. When arbitrarily many routes from a node to the base station are permitted, optimal routes may be found as the solution to a well-known linear program. We present an evolutionary algorithm that finds good routes when each node is allowed only a small number of paths to the base station. On a real network deployed in the Victoria & Albert Museum, London, these solutions, using only three paths per node, are able to achieve minimum lifetimes of over 99% of the optimum linear program solution's time to first sensor battery failure.

Keywords

Mesh networks, shortest path, evolutionary algorithms, multi-objective optimisation, network lifetime optimisation

1 Introduction

Wireless Sensor Networks (WSNs) consist of autonomous devices distributed over a wide area that are able to sense and periodically report environmental parameters such as temperature or humidity. They are extensively used for remote monitoring, especially in industrial, regulatory and heritage applications. Many applications require sensors to be placed far away from easy access to mains power, so battery-powered sensors are popular and are often necessary in such situations. However, it is also desirable that sensors can be left unattended, without battery replacement for as long as possible. In this paper we therefore investigate the use of hybrid evolutionary approaches to find routing schemes for *mesh* wireless sensor networks that optimally preserve the life of the network and promote energy efficiency.

©200X by the Massachusetts Institute of Technology

Many existing commercial products use an *ad hoc* topology for WSNs. Generally, these *ad hoc* networks are point-to-point networks, in which each sensor reports directly to a central base station. Despite being a low power solution, a major limitation is limited network range as each sensor *node* must be able to communicate directly with the base station. The rigid network structure also means that networks are unable to cope with changing conditions in a dynamic radio environment, which may occur as buildings are modified or furniture is moved. As a consequence sometimes multiple base stations are required to cover a single site.

As a solution, in recent years, mesh network topologies, where data is relayed from node to node *en route* to the base station, have grown in popularity. In addition to extending the range that can be covered by using multiple hops (that is multiple node-to-node relays) mesh networks provide the opportunity of using alternative routes, thus increasing resilience to radio environment changes. However, mesh networks are expensive in terms of energy consumption due to a higher overhead at each node for additional activities, namely relaying messages for other nodes and, in systems with distributed planning, calculating new routes. These additional activities can be severely detrimental to the overall life of the network, reducing the time before it requires servicing and battery replacement. It is desirable to minimise the energy consumption in the network, which is equivalent to maximising average lifetime (Rodoplu and Meng, 1999). However, as Chang and Tassiulas (2004) indicated, using minimum energy routes throughout the lifetime of a network can be detrimental to a group of nodes that relays most routes. Hence, in addition to improving average lifetime, it will usually be important to maximise the time before the battery of the shortest-lived node is exhausted. This problem is called the maximum lifetime routing problem (Chang and Tassiulas, 2004; Madan and Lall, 2006). Routing optimisation in mesh networks therefore requires consideration of the trade-off between individual and system-wide battery lifetimes within the network.

Both efficient energy usage and maximum lifetime routing have attracted much research interest in recent years. These approaches can be divided into two groups: distributed and centralised (Kulkarni et al., 2011). In distributed approaches, the responsibility for routing is distributed across the constituent nodes, i.e. the nodes are able to make decisions on routing locally. On the other hand, in centralised approaches, a centrally calculated route is broadcast to participating nodes.

Distributed approaches can provide good performance, and even optimal solutions. For instance, the distributed approach described by Rodoplu and Meng (1999) is guaranteed to converge to the minimum energy topology in a strongly connected network where the communication links are not vulnerable to environmental changes. In addition, Madan and Lall (2006) described a distributed approach that can locate the optimal routing scheme using a sub-gradient algorithm to solve the convex optimisation problem presented by maximum lifetime routing. Also, some heuristic approaches, such as reinforcement learning (Förster, 2007) and swarm intelligence (Bashyal and Venayagamoorthy, 2007), can be applied in a distributed fashion, although these only approximate the optimal solutions. An important consideration with distributed approaches is that nodes require sufficient computational power and storage to collect and store information regarding local connectivity and compute the best routes based on available information. In comparison, centralised approaches, which mostly incorporate variants of heuristic algorithms (Chang and Tassiulas, 2004; Xue et al., 2006; Islam and Hussain, 2006; Yetgin et al., 2012) require lower computational power and storage at the nodes, as most of the computation and storage is conducted by the central base station. Nonetheless, there is a system-wide overhead incurred in gathering connectivity information and broadcasting routing information. In this paper we consider very low powered nodes, each of which has limited computational power and storage. Routes are therefore computed at a mains-powered base station.

Most current evolutionary algorithm (EA) based energy-aware centralised systems consider energy expense. This is the case even for multi-objective routing optimisation, where energy expense is optimised with additional objectives describing different factors, such as quality of service, bandwidth, packet loss ratio, etc. (Xue et al., 2006; Yetgin et al., 2012). However, optimising the overall energy expenditure of a network may be detrimental to the overall performance of the network, because often the goal is to prolong the lifetime of the network before any battery needs replacing. Merely reducing the overall energy expenditure may place a large burden on a few nodes, resulting in the rapid exhaustion of their batteries. We therefore seek to optimise the lifetime of network nodes by modelling the charge held in their batteries and the energy expenditure at *each* node. Islam and Hussain (2006) and Kamath and Nasipuri (2011) have considered maximising only the minimum remaining lifetime among nodes. Such approaches can improve the individual node specific energy state, but can be sub-optimal from system-wide perspective. We therefore seek to find the optimal trade-off between local and network-wide battery lifetimes.

In previous work (Rahat et al., 2014) we investigated a centralised approach using multi-objective routing optimisation with a view to locating efficient routes that approximate the optimal trade-off between minimum lifetime and average lifetime among nodes in the system. We also described how our system can cope with failure and permits self-healing in a dynamic radio environment. Our results indicated that using multiple routing schemes may allow the minimum lifetime of the system to be extended. In a mesh network where each node is able to route its messages to the base station via a number of different routes Chang and Tassiulas (2004) have shown that a linear programming (LP) problem may be solved to obtain the proportions of messages that should be sent by each different route so as to achieve the maximum minimum lifetime. However, in very low powered battery networks the requirement to use many routes places an undue burden on the storage and computation that must be performed at each node. We therefore consider a routing scheme in which each node uses only a single route to the base station, but at a few discrete times the routing scheme for the entire network can be changed. We present a hybrid multi-objective evolutionary algorithm to maximise the minimum lifetime of any node in the system and the average lifetime of all nodes in the system. This algorithm is shown to achieve minimum lifetimes very close to the LP solution obtained by Chang and Tassiulas (2004) (which cannot directly be used here, due to its use of many routes per node). We also show how a different linear program may be used to find the optimum proportion of time for which each routing scheme should be used, thus obviating the need for an additional evolutionary search for the routes' time periods.

The number of potential solutions suffers from a combinatorial explosion as the number of nodes grows. To combat this we use two methods of pruning the search space, one based on the k-shortest paths from a node to the base station and another that utilises the solution to the maximum minimum lifetime linear program to find potentially useful routes.

In section 2 we describe the model of the wireless sensor network. In section 3 we present the maximum minimum lifetime LP problem and discuss why maximising the average lifetime cannot be formulated as an LP problem. In section 4 we present

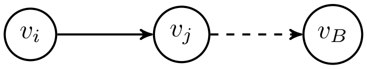


Figure 1: Route $S_i = \langle v_i, v_j, \ldots, v_B \rangle$ from v_i via v_j to the base station v_B.

search space pruning methods incorporating k-shortest paths and a graph reduction method using the LP solution to the maximum minimum lifetime problem. These are combined in section 5 where we present a hybrid evolutionary routing optimisation strategy. Based on this, in section 6, we demonstrate and discuss our findings in a real network deployed in the Victoria & Albert Museum, London. In section 7 we discuss how minimum lifetime of the system can be improved by using multiple routing schemes in an optimal time-shared manner. Finally, conclusions are presented in section 8.

2 System Model and Pareto Optimality

In this section we model the WSN and derive a multi-objective problem in order to investigate how different routing schemes affect the trade-off between individual and system average lifetimes.

A WSN is represented as a network graph, $G = \{V, E\}$, where V is a finite set of n sensor nodes v_i plus a base station node, v_B, and E is the finite set of m edges (Cormen et al., 2001) describing which nodes each node can communicate with. Each node must send messages to the base station, perhaps by relaying a message through one or more other nodes. It is not acceptable for nodes to be disconnected from the network (i.e. to be unable to directly or indirectly route a message to the base station). In practical situations where this occurs, it is the role of the network engineer to distribute additional 'repeater' nodes to bridge communication gaps, or to use nodes which can transmit at greater strengths. Once all nodes can pass on a message (directly or indirectly) to the base station, the routing optimisation stage (the focus of this paper) may be undertaken. In our scheme each node reports its status using a single route once every reporting cycle (e.g. once each minute).

As illustrated in Figure 1, a route from node v_i to the base station v_B is described by the sequence, $S_i = \langle v_i, v_j, \ldots, v_B \rangle$. We denote by $S_i[p]$ the pth element of the route S_i. A *routing scheme* R is a set of routes, one for each node in the network, to the base station:

$$R = \{S_1, S_2, \ldots, S_n\}. \tag{1}$$

An initial mapping phase, preceding optimisation, is used to discover with which other nodes the node v_i can communicate. We call this map the connectivity map. Communications may take place using a variety of baud rates and powers, so we assume that the most energy efficient baud rate and power combination has been discovered for each pair of nodes that can communicate; generally low power and high baud rates are most efficient in our system, but the optimisation does not rely on this. Also we assume that communication is reliable at the chosen baud rate and power combination.

The energy required to transmit a message from v_i to the base station is the sum of the energies required to transmit a message between each of the nodes comprising the route:

$$H_i = \sum_{p=1}^{l-1} e_{S_i[p], S_i[p+1]} \tag{2}$$

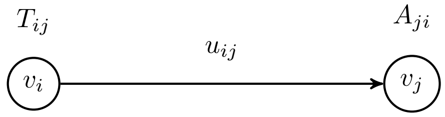


Figure 2: A transmission link between node v_i and v_j. The associated energy costs to send a message from v_i to v_j are: T_{ij} at node v_i and A_{ji} at node v_j. The number of times this edge is used by different routes, i.e. the edge utilisation, is u_{ij}.

where l is the length of the route and e_{jk} is the energy required to transmit a message from v_j to v_k. Note that this generally involves energy expenditure at both the transmitting node and the receiving node, and will also involve expenditures for transmitting an acknowledgement. As noted above, we assume that the communication is reliable, but if an acknowledgement is not received from the receiver the message is resent; this additional expense is not modelled, but if a link becomes unreliable, the routing is re-optimised.

In many routing optimisation problems, such as shortest path problems, minimising a route's overall cost is desirable. The overall cost is found by summing the costs associated with each edge in the route. There are many well-known methods for minimising such costs, e.g. (Eppstein, 1998). In this problem, however, we focus on the costs expended at the *nodes* themselves, rather than the edge costs. This is because it is energy expended at the nodes that depletes charge in the batteries and thus governs the lifetime of a node.

Let T_{ij} be the energy (charge) required at node v_i to send a message to v_j (Figure 2) and let A_{ki} be the energy required to receive a message from v_k at v_i. Also let the edge utilisation u_{ij} for a directed edge between v_i and v_j be the number of times in one reporting cycle that the particular edge is used by the routing scheme for a transmission between the associated nodes. Then in one reporting cycle, node v_i receives messages from nodes with indices in the set $\mathcal{I}_i$ and sends data (including own data) to nodes with indices $\mathcal{O}_i$; the associated energy expense is

$$C_i = \sum_{k \in \mathcal{I}_i} u_{ki} A_{ki} + \sum_{j \in \mathcal{O}_i} u_{ij} T_{ij}. \tag{3}$$

Clearly $\sum_i^n C_i = \sum_i^n H_i$ is equal to the energy cost across the whole network of sending a message from each node.

In order to calculate the lifetime remaining due to a routing scheme we require additional intrinsic information about the nodes, namely the charge q_i remaining in the battery and the quiescent energy consumption per reporting cycle B_i due to constant micro-controller operation, sensor measurements, running an on-board display, etc. The life of the current node therefore is modelled as

$$L_i = \frac{q_i}{(B_i + C_i)N} \tag{4}$$

where N is the number of reporting cycles per unit time. We emphasise that $L_i \equiv L_i(R)$ is a function of all the routes which utilise v_i, and C_i is calculated from R.

Our goal is to prolong the average life of the network, that is to minimise the total energy consumed, and to maximise the time before any individual node requires its battery to be recharged or changed. We therefore arrive at the two objective problem:

$$\text{Maximise} \quad f_1(R) = \frac{1}{n}\sum_{i=1}^{n} L_i(R). \tag{5}$$

$$\text{Maximise} \quad f_2(R) = \min_{i\in[1,n]} L_i(R). \tag{6}$$

In addition, it may be important to prolong the lifetime of one or more nodes v_i for $i \in \mathcal{U}$, because, for example, they are particularly inaccessible. In this case the two-objective problem is augmented with a third objective:

$$\text{Maximise} \quad f_3(R) = \min_{i\in\mathcal{U}} L_i(R). \tag{7}$$

The energy efficiency objective in (5) will ensure paths with the least energy consumption are selected irrespective of the load imposed on the nodes. As a consequence, certain nodes in the network may end up relaying most of the traffic (depending on the network structure). This is in conflict with the minimum lifetime objectives in (6) and (7), as to maximise these objectives it is better to distribute some load away from the nodes relaying most traffic. Additionally, (6) and (7) will be in conflict when the minimum lifetime node $v_m \notin \mathcal{U}$ and $\mathcal{U} \neq V$.

Solving this multi-objective problem may result in multiple solutions, as opposed to a single solution for single objective optimisation. In this case, there exists a set of solutions which are Pareto optimal; that is, there are no other feasible solutions available that improve performance on one objective, without a simultaneous decrease in at least in one other objective (see, for example, Coello Coello et al., 2001).

The dominance criterion is used to locate such solutions in the search space. The dominance criterion from a routing optimisation perspective is described as follows. In a multi-objective problem with M objectives, a routing scheme, R', is said to dominate another routing scheme, R, denoted $R' \succ R$, iff

$$\begin{aligned} &f_m(R') \geq f_m(R)\; \forall m = 1, 2, \ldots, M, \text{ and} \\ &f_m(R') > f_m(R) \text{ for some } m. \end{aligned} \tag{8}$$

Hence, we seek the maximal set of feasible routes which are mutually non-dominating, which is known as the Pareto set.

Note that an alternative approach is to combine multiple objectives in a single objective function as a weighted sum of the constituent objectives. A set of weights then represents the relative importance of the underlying objectives as set by the decision maker, and this technique can only find a solution relating to a particular trade-off in a single optimisation run (Coello Coello et al., 2001). However, Das and Dennis (1997) showed that performing multiple optimisation runs using evenly distributed weights fails to locate an even distribution of solutions across all parts of the Pareto front, even if the front is convex. Therefore, we consider these objectives as part of a multi-objective problem.

3 Maximum Lifetime Routing and Energy Efficiency

Maximising the time before at least one node exhausts its power source is often referred to as the maximum lifetime routing problem (Chang and Tassiulas, 2004; Madan and Lall, 2006). Chang and Tassiulas have identified the maximum lifetime routing as a linear programming problem. This is particularly useful as this LP problem can be solved in polynomial time. We use this solution in our hybrid evolutionary method, not only as a yardstick for our approach, but also as a means of directing the search. In this section, we present the LP formulation for maximum lifetime routing and discuss why the problem of maximising the average lifetime cannot be formulated as a linear program.

The LP problem formulated by Chang and Tassiulas (2004) incorporates multiple sets of source-destination pairs, and can be generalised when these pairs are active at different times with variable reporting rates in a multi-commodity setting. In this paper, we consider a special case of their formulation, where all nodes send the same amount of data periodically at fixed intervals, e.g. a message every minute. This scenario is most common in industrial applications, especially for constant monitoring of locations. Also, we extend the approach with the inclusion of quiescent consumption at nodes as a practical consideration.

At the heart of the formulation of the LP problem lies the concept of network flow conservation: the outgoing flows from a node v_i must be equal to the sum of the incoming flows from other nodes and the flow generated at v_i. The network flow effectively indicates edge utilisations, i.e. how many times a particular edge has been used in a routing scheme. Hence, the flow conservation at v_i can be written in terms of edge utilisations.

$$\sum_{k \in \mathcal{I}_i} u_{ki} + U_i = \sum_{j \in \mathcal{O}_i} u_{ij}, \tag{9}$$

where u_{ij} is the edge utilisation of the link from v_i to v_j and U_i is the flow generated at v_i, namely the data generated in each reporting cycle at v_i; the recipient node indices belong to set $\mathcal{O}_i$ and nodes transmitting to v_i have indices in the set $\mathcal{I}_i$.

A mathematical programming formulation for the maximum lifetime routing can be derived in the following way. The objective is:

$$\max \left(\min_{i=[1,n]} L_i \right), \tag{10}$$

which is the second objective of our multi-objective problem as presented in (6). The associated constraints of this problem for $v_i \in V$ are:

$$u_{ij} \geq 0, \tag{11}$$

$$NL_iB_i + \sum_{j \in \mathcal{O}_i} NL_i u_{ij} T_{ij} + \sum_{k \in \mathcal{I}_i} NL_i u_{ki} A_{ki} \leq q_i, \tag{12}$$

$$\sum_{k \in \mathcal{I}_i} u_{ki} + U_i = \sum_{j \in \mathcal{O}_i} u_{ij}. \tag{13}$$

Here the first constraint (11) ensures that edge utilisations are non-negative, the second constraint (12) expresses the fact that the energy usage of a node cannot exceed the remaining charge at the node, and the final constraint (13) represents the flow conservation in the network.

The second constraint (12) is derived from the definition of lifetime as provided in (3) and (4). Note that this expression is non-linear in u_{ij}, which means in general that the feasible region is non-convex and that methods guaranteed to find the optimum in polynomial time are not available (Boyd and Vandenberghe, 2004); furthermore, the max-min objective function (10) is non-smooth (Zang, 1980).

In the minimum lifetime case: $L_i \geq \min_{i=[1,n]}(L_i) = \mathcal{L}$, so L_i may be replaced with $\mathcal{L}$. Chang and Tassiulas (2004) therefore recast the problem in terms of $\hat{u}_{ij} = \mathcal{L}u_{ij}$ to obtain an LP problem that can be solved in polynomial time (Boyd and Vandenberghe, 2004). The LP problem has the following objective:

$$\max(\mathcal{L}), \tag{14}$$

subject to the following modified constraints:

$$\hat{u}_{ij} \geq 0, \tag{15}$$

$$N\mathcal{L}B_i + \sum_{j\in\mathcal{O}_i} N\hat{u}_{ij}T_{ij} + \sum_{k\in\mathcal{I}_i} N\hat{u}_{ki}A_{ki} \leq q_i, \tag{16}$$

$$\sum_{k\in\mathcal{I}_i} \hat{u}_{ki} + \mathcal{L}U_i = \sum_{j\in\mathcal{O}_i} \hat{u}_{ij}. \tag{17}$$

This LP problem may be solved to obtain the minimum lifetime for the system, $\mathcal{L}$, and a set of compound edge utilisations $\hat{u}_{ij}$.

The energy efficiency of WSN system is often viewed in terms of the average lifetime of the nodes. Maximising this ensures best possible usage of the energy available in the network as a whole (Rodoplu and Meng, 1999). Similar to the minimum lifetime problem, the objective is a linear function of the edge utilisations (5) with the same constraints (equations (11), (12), and (13)) as for the minimum lifetime problem. Unfortunately, in this case the multiple quadratic constraints cannot be reformulated to obtain an LP problem meaning that efficient polynomial time methods cannot be employed. In addition, when the number of routes from a node to the base station is limited the Chang and Tassiulas (2004) method cannot be used to obtain an LP method for solving the minimum lifetime solution. We therefore turn to evolutionary methods to locate an approximate Pareto front describing the trade-off between these objectives.

4 Search Space Pruning

The multi-objective optimisation problem described in section 2 is a combinatorial optimisation problem with, for practical WSNs, a vast number of potential solutions. The number of possible routing schemes, i.e. the search space size, depends on the number of available routes for each node in the system. For instance, in a network with n nodes excluding the base station let the number of available loopless paths from v_i to v_B be a_i. In this case, the number of possible routing schemes, i.e. the number of combinations of routes for individual nodes that can build the routing scheme, is:

$$Z = \prod_{i=1}^{n} a_i. \tag{18}$$

It is crucial for practical implementations that the optimisation process is fast. A way to improve the speed of optimisation is to sensibly prune the search space, while retaining important potential solutions. In this section, we describe two methods of pruning the search space: k-shortest path pruning and a max-min lifetime pruning, based on the maximum minimum lifetime solution, and discuss how they are used in approximating the Pareto set.

4.1 k-shortest Path Pruning

In order to combat the potential growth in the size of the search space as the number of nodes increases we limit the number of potential routes available to each node. More specifically, we limit the search to the space defined by the k-shortest paths for each node, where the metric defining the distance between nodes (the weight of the edges) is described below. This reflects our intuition that short paths to the base station are most likely to be energy efficient. We select from among several shortest path routes for each node because, if each node were to utilise its shortest path, nodes that occur in many of the 1-shortest paths would be disproportionately burdened.

Algorithms for discovering the shortest path between two nodes in a weighted graph are well known and the shortest path can be found in $O(n \log n)$ time (Yen, 1971; Eppstein, 1998; Brander and Sinclair, 1995). However, as we noted above, the energy costs in this problem are associated with the nodes themselves rather than with the edges. We therefore weight the edges in the network graph to associate the energy cost at the nodes with the edges connecting them. Consider the nodes v_i and v_j. We define the weight of the edge between them as:

$$w_{ij} = \frac{e_{ij}}{q_i} + \frac{e_{ji}}{q_j}, \tag{19}$$

where, as above, e_{ij} is the energy required to transmit a message from v_i to v_j, and q_i and q_j are the battery charges. It is expected that $e_{ij} = e_{ji}$. This edge weighting models the fact that a high transmission cost can be borne by nodes with a high battery charge, but transmission is relatively expensive for nodes with low battery charge because each transmission will make a larger fractional depletion of the charge. Likewise, if a node is connected to mains power then transmissions are free, which is modelled by setting $q_i \to \infty$. Note that $q_i / \sum_j e_{ij}$ is an estimate of the lifetime of v_i. We call the cost of a routing scheme calculated using the weights w_{ij} the *composite cost*.

As we require diversity in the search space and the possibility of load balancing among nodes, we propose to evolve solutions from among the k-shortest paths for each node calculated with the composite cost (19). A number of algorithms are available for computing the k-shortest paths; see for example (Yen, 1971; Eppstein, 1998). In our implementation we have used Eppstein's algorithm modified to produce only simple or loopless paths, which matches the best upper-bound time complexity for finding k-shortest simple paths (Eppstein, 1998). We denote the mth shortest route found for node v_i by S_i^m, for $m = 1, \ldots, k$. For a chosen k, we build a library $\mathcal{P} = \langle \{S_1^m\}_{m=1}^k, \{S_2^m\}_{m=1}^k, \ldots, \{S_n^m\}_{m=1}^k \rangle$ of paths, where each node has no more than k paths. This is because the number of loopless paths to the base station, v_B, for a node may be less than k, especially if it is close to v_B. A complete routing scheme R is then represented by a path for each node selected from one of the paths available in the library. Using this technique the reduced search space of possible solutions, Ω, is no larger than k^n.

We also investigated the correlation between the composite cost and the objectives for 1,000 randomly chosen routing schemes with random charge levels q_i at nodes for a real network at the Victoria & Albert museum, which we describe in more detail below (Rahat et al., 2014). We found that there is a fairly strong negative relationship between the average battery life $f_1(R)$ and the composite cost (correlation coefficient -0.71). This indicates that routing schemes selected with low composite cost via the k-shortest paths algorithm are likely to have long average lives. On the other hand, there was a very weak relationship between the minimum lifetime and composite cost. This was unsurprising since the minimum lifetime depends on a *single* node in the network and is highly dependent on the interaction between routes.

4.2 Max-Min Lifetime Pruning

The correlation between composite edge cost and average battery lifetime, but poor correlation between composite edge cost and minimum battery life means that although the k-shortest paths pruning retains good average lifetime routes it may discard routes that would yield a long minimum lifetime. To obtain good candidate routes for long minimum lifetimes we solve the linear program to obtain the edge utilisations u_{ij} which maximise the minimum lifetime. We then consider the subgraph $G' \subset G$ obtained by deleting from G all the edges e_{ij} for which $u_{ij} = 0$. This subgraph G' is then used as the basis for generating a library $\mathcal{P}'$ of k-shortest paths with Eppstein's algorithm. The use of G' prunes the size of the search space, but retains routes that are good for prolonging the minimum lifetime. We call this reduced search space Ω'.

The purpose of using such search space pruning is to retain routes from which to build good solutions. Here each method has its own specific goal: k-shortest path pruning for retaining solutions with better average lifetime and max-min lifetime pruning for retaining solutions with better minimum lifetime. We can then utilise the modified search space to rapidly obtain a sensible approximation of the optimal Pareto set. Although $G' \subset G$, the k-shortest paths derived from G' may be different from those derived from G. As a consequence, using the k-shortest path pruning may result in the search spaces Ω and Ω' being quite dissimilar if not completely disjoint.

5 Hybrid Evolutionary Approach to Routing Optimisation

The main focus of our approach is the rapid approximation of the trade-off front between minimum lifetime and average lifetime from infrequent maintenance and energy efficiency perspectives respectively. In this section, we describe how we use hybrid evolutionary approaches in a multi-objective evolutionary algorithm to locate an approximately optimal set of routes.

As described in the previous section, we prune the search space in two ways to obtain path libraries, $\mathcal{P}$ and $\mathcal{P}'$, of potential routes for each node in the network. We represent a solution as a vector of indices into these libraries, one index for each node, so that a complete solution R describes a route for each node to the base station. One optimisation strategy would be to select a route for each node from $\mathcal{P} \oplus \mathcal{P}'$. However, as we demonstrate below, it is more effective to optimise in two stages. In the first stage, we perform two separate optimisations, one selecting routes from $\mathcal{P}$ and the other selecting routes from $\mathcal{P}'$. In the second phase routes are selected from $\mathcal{P} \oplus \mathcal{P}'$. This phase is initialised from the non-dominated archives resulting from the two first stage optimisations.

Optimisations for both stages use a genetic algorithm, with an unconstrained Pareto archive to reap the benefits of better convergence properties (Fieldsend et al., 2003). Algorithm 1 describes this multi-objective optimisation process in more detail.

Solutions, R, are represented by vectors of n integers; the ith element of the solution indexes one of the k-shortest paths found for node v_i. In the initialisation step, we generate at random a population from the pruned search space, Ω. (The search space is Ω' when using the max-min pruning, but we describe the algorithm in terms of Ω.) This population comprises randomly-chosen routing schemes where the member routes are chosen from the k-shortest paths for each node; thus for node v_i the shortest paths are selected from $\{S_i^m\}_{m=1}^k$. Also, we include in the initial population the routing scheme

Algorithm 1 Multi-objective routing optimisation of a battery powered mesh network.

```
Inputs
$\mathcal{P}$ : Library of paths for each node
$T$ : Number of iterations
$s$ : Size of initial archive
$\mu$ : Perturbation rate
$c$ : Crossover rate

Steps
$A \leftarrow \text{InitialiseArchive}(\mathcal{P}, s)$                ▷ Initialise random archive
for $i = 1 \rightarrow T$ do
    $\{R_1, R_2\} \leftarrow \text{Select}(A)$                   ▷ Select two parent solutions
    $R' \leftarrow \text{CrossOver}(R_1, R_2, c)$
    $R'' \leftarrow \text{Perturb}(R', \mu, \mathcal{P})$                          ▷ Mutation
    $A \leftarrow \text{NonDominated}(A \cup R'')$                     ▷ Update archive
end for
return $A$                                      ▷ Approximation of the Pareto set
```

which uses the first shortest route for each node. The initial archive of non-dominated solutions A is the maximal non-dominated subset of this random population. At any step of the evolution, A is the current approximation of the Pareto set within the particular chosen search space.

During the evolution process, we randomly select two routing schemes. These parent routing schemes are then crossed-over to yield a single offspring R': the route for each node v_i is selected either the first or second parent with probability c and $1 - c$ respectively, independently of the other nodes. This child is then perturbed by choosing new routes (uniformly at random from $\mathcal{P}$) for a proportion μ of the nodes in the network. The perturbed child, R'', is compared against the members in the archive: if it is not dominated by any of them then it enters the archive and any elements of A which are dominated by the new solution are removed from A. In this fashion only the non-dominated routing schemes are preserved in the archive and the archive can only approach the true Pareto set. The process of evolution continues for a fixed number of episodes. Alternatively, another termination criterion, such as a specified minimum dominated hypervolume (Zitzler, 1999), may be employed.

We denote the estimated Pareto sets resulting from the separate optimisations using $\mathcal{P}$ and $\mathcal{P}'$ as A and A' respectively. As illustrated below, the archive A resulting from optimisation using $\mathcal{P}$ tends to favour solutions with long average lifetimes, whereas A' resulting from optimisation using $\mathcal{P}'$ tends to yield solutions with long minimum lifetimes. However, solutions with intermediate average and minimum lifetimes may be scarce in $A \cup A'$. The second phase of the optimisation remedies this by employing the same optimisation algorithm, but now selecting paths from $\mathcal{P} \oplus \mathcal{P}'$. This optimisation is initialised from the maximal non-dominated subset of $A \cup A'$; that is $A''_{init} = \text{NonDominated}(A \cup A')$, where $\text{NonDominated}(\cdot)$ is the function that returns the maximal subset of non-dominated elements of X:

$$\text{NonDominated}(X) = \{R \in X \mid \nexists (R' \in X \wedge R' \prec R)\}. \tag{20}$$

We denote the archive resulting from the second optimisation by A''.

The overall hybrid evolutionary approach can be summarised as follows:

1. Gather the connectivity map, and construct the network graph G.
2. Solve the LP problem for maximum lifetime routing (10) to find the optimal minimum lifetime $\mathcal{L}$ and edge utilisations u_{ij}.
3. Search space pruning.
   (a) Use u_{ij} to generate the reduced graph G' by selecting edges where $u_{ij} > 0$.
   (b) Apply k-shortest path pruning to G and G' creating path libraries $\mathcal{P}$ and $\mathcal{P}'$ respectively.
4. First stage of routing optimisation. Apply Algorithm 1 using $\mathcal{P}$ and $\mathcal{P}'$ separately. This results in archives A and A', which are the approximations of the Pareto set in the relevant search space.
5. Initialise the second stage optimisation with $A''_{init} = \mathrm{NonDominated}(A \cup A')$.
6. Second stage of routing optimisation: Apply Algorithm 1 using $\mathcal{P} \oplus \mathcal{P}'$. On termination, we have the estimated Pareto set A''.

Once the evolution process is finished, the decision maker may manually choose the operating point using the final approximation of the Pareto front A''. The chosen routing scheme is sent to the nodes via the base station and becomes the active data reporting scheme.

6 Illustration

A real network deployed in the Victoria & Albert Museum, London, is used to illustrate the proposed approach.[1] In a controlled environment over a vast area, such as Victoria & Albert Museum, it is essential to monitor temperature and humidity in galleries and display cases for the preservation of the artefacts. Compared to wired networks, battery powered WSNs carry huge advantages in deployment cost and flexibility.

The network incorporated 30 sensor nodes and a base station, spanning five floors within an approximate area of 35,000 m^2. The thick, solid walls provide a challenging radio environment whose characteristics vary with the passage of visitors through the galleries.

6.1 Baseline Optimisation

The connectivity map G was built in an initial mapping phase in which nodes *pinged* each other using a range of baud rates and powers to discover the minimum energy configuration for communication between those nodes within radio range of each other.

Solving the maximum minimum lifetime LP problem results in a maximum minimum lifetime of 1.38 years, together with a set of edge utilisations. These edge utilisations were used to calculate the average lifetime of the system, which is 1.69 years. They also enable the extraction of the sub-graph G'. We emphasise again that this maximum minimum lifetime is only achievable through the use of multiple routes from

[1] Data on the network to allow comparison with this work can be found at `http://emps.exeter.ac.uk/computer-science/wsn/`

each node to the base station, whereas we are constrained to seek a routing in which each node uses only a single route to communicate with the base station. Clearly, the requirement to use a single route means that battery loadings cannot be shared optimally so that a single routing scheme will have a worse (shorter) maximum lifetime than that given by linear programming.

Applying subspace pruning using $k = 10$ shortest paths results in libraries $\mathcal{P}$ and $\mathcal{P}'$ from each of which 10^{30} solutions could be constructed. In each search space, the initial population was built with 100 randomly chosen solutions from the relevant subspace together with the solution consisting of the shortest route for each node. The initial archive was then found by extracting the maximal set of non-dominated solutions in this population.

In the evolutionary optimisation we used $\mu = c = 0.1$ as the perturbation and crossover rates; these rates were chosen after a short empirical study on simulated networks, but the performance of the optimiser is relatively insensitive to their precise values. Using the dominated hypervolume measure (Zitzler, 1999), the Pareto sets for the first stage optimisation were well converged after 150,000 epochs (Figure 6). The second stage optimisation used the same μ and c as the first stage. This evolution was run for 500,000 epochs, resulting in the final approximation of the Pareto set A''.

A single core Python implementation takes about 2 minutes to complete each of the initial 150,000 iterations on a 2.5 GHz machine. The 500,000 iterations in the second stage takes about 4 minutes to complete. Consequently, the routes for new installations can be found readily, making this approach particularly feasible as part of a real system.

Figure 3 shows the estimated Pareto fronts generated during the initial stage of optimisation. As the figure suggests, in both search spaces, Ω and Ω', there is a wide range of routes which trade-off the average lifetime of the network against the time before any single node exhausts its battery. We note that the optimisation has resulted in routing schemes that are substantially better, in terms of both average and minimum lifetime, than routing schemes utilising randomly chosen routes from the 10 shortest paths in both Ω and Ω'. Indeed, for both search spaces, the routing schemes with the longest minimum lifetime have a better average lifetime than any of the random solutions for that search space. Interestingly, the shortest paths only solutions R_c and R'_c are very close to the estimated Pareto front in both spaces. We include R_c and R'_c in the initial archives for the relevant search spaces.

We note that for the Victoria & Albert Museum network 13 of the 30 nodes have a possible path which appears in both $\mathcal{P}$ and $\mathcal{P}'$, although no node has more than 3 paths common to both libraries. Since 17 of the nodes therefore have no paths appearing in both libraries, it is not possible to construct a complete routing scheme, R, that exists in both libraries and the search spaces Ω and Ω' are completely disjoint. Also the fronts occupy different parts of the objective space. Many of the solutions in A' (resulting from the max-min pruning) are dominated by solutions in A, which resulted from k-shortest path pruning of the full graph G. Nonetheless, the utility of using the min-max pruning is shown by the 11 solutions that achieve longer minimum lifetimes than any solution found in Ω.

Combining A and A' leaves a large gap in the estimated Pareto front for intermediate average lifetimes. To remedy this, the second stage optimisation is initialised from $\mathrm{NonDominated}(A \cup A')$ and the union of the two pruned search spaces is available for constructing new solutions. The result of this second optimisation is shown in Figure 4.

As can be seen from Figure 4, A'' is very close to A in the high average lifetime

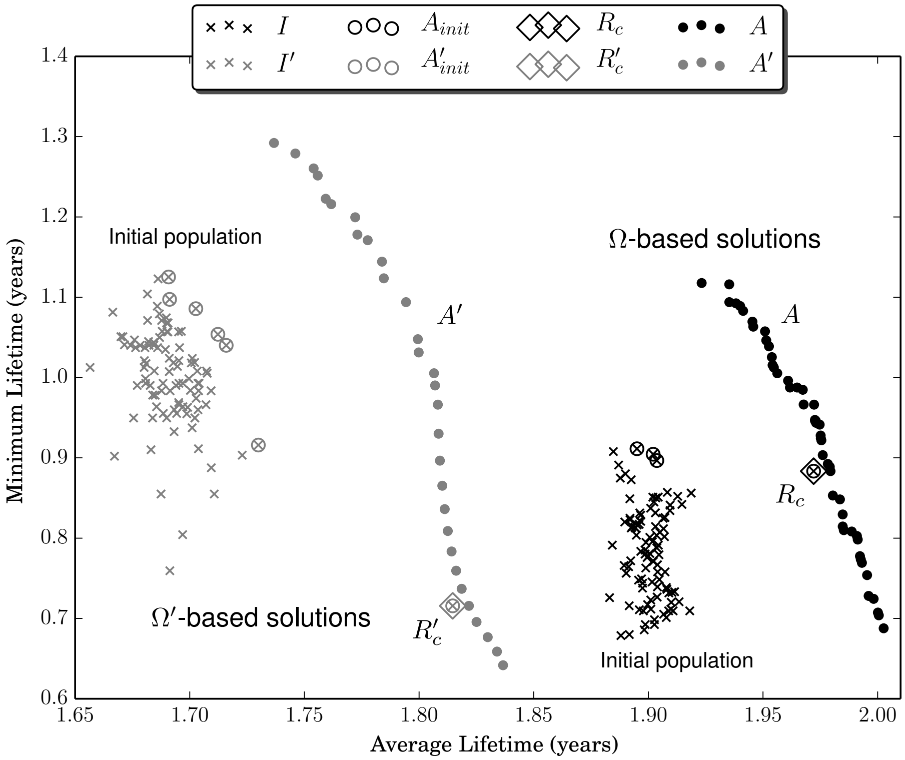


Figure 3: First stage optimisations using two pruned search spaces Ω and Ω' for the Victoria & Albert museum network. Initial randomly generated solutions and the optimised fronts are shown. Grey coloured symbols represent solutions for Ω', while black indicates solutions for Ω. Initial populations (I in Ω and I' in Ω') are shown with crosses; ringed crossed show the non-dominated initial archives, A_{init} and A'_{init}. The shortest paths only solutions (R_c in Ω, and R'_c in Ω') are indicated with diamonds.

region, but combining paths from $\mathcal{P}$ and $\mathcal{P}'$ has resulted in solutions being found with good minimum lifetimes and better average lifetimes than those in A'. The linear programming solution (that solves for many routes per node rather than a single route) is marked by a black triangle in the figure and this upper bound on the minimum lifetime, $\mathcal{L} = 1.38$ years, is indicated by the horizontal dashed line. The best minimum lifetime solution when nodes are constrained to use a single route found by the evolutionary optimiser has a minimum lifetime of 1.29 years. This solution improves the shortest-path only routing scheme by approximately 5 months in minimum lifetime at an expense of 2 months in average lifetime. In contrast, the solution with best average lifetime improves the shortest-path only solution by 7 days in average lifetime, while the minimum lifetime is worse by 2 months, which is a rather small gain at a large expense.

Figure 5 shows a comparison between the two-stage approach and a single-stage approach in which solutions may be constructed from $\mathcal{P} \oplus \mathcal{P}'$ throughout the entire optimisation. As can be seen from Figure 5, the estimated Pareto fronts resulting from the two strategies mostly overlap, except at the extreme maximum minimum lifetime/

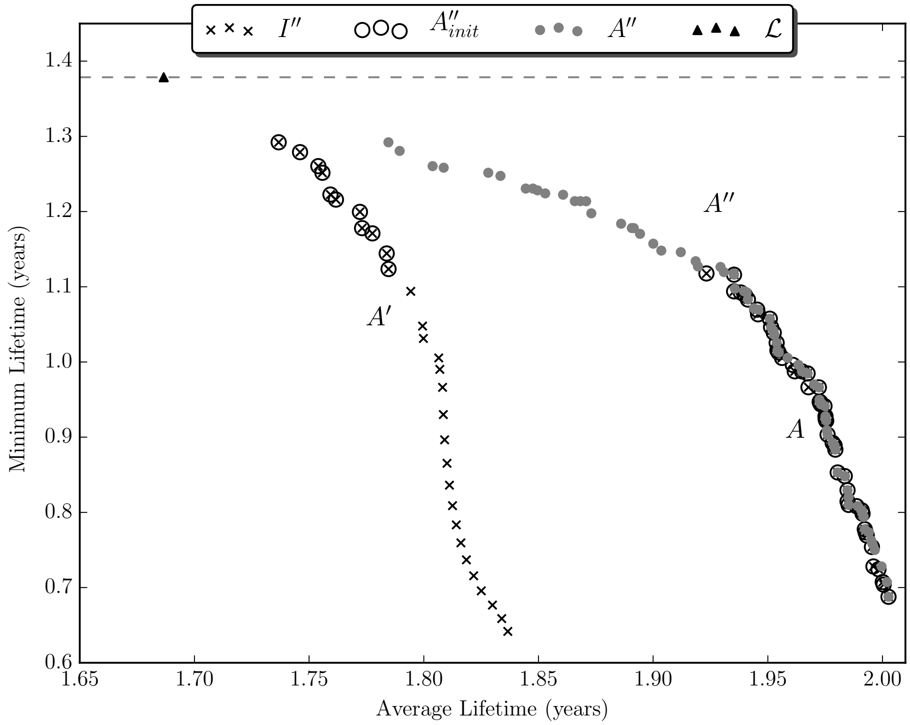


Figure 4: Final stage of optimisation. The initial population (represented with crosses) is generated by combining the archives from the previous stage; c.f. Figure 3. The non-dominated elements of $A \cup A'$ are ringed, and form the initial archive of the second stage routing optimisation problem (A''_{init}). Solid grey dots show the final approximation of the Pareto set (A''). The maximum minimum lifetime solution if nodes were allowed to use multiple routes rather than a single route is indicated by the black triangle. The horizontal dashed line through $\mathcal{L}$ is an upper bound to the minimum lifetime.

minimum average lifetime end. Here, the strategy of evolving solutions using a path library that favours good minimum lifetime solutions has paid off, yielding additional solutions with long minimum lifetimes. To generate this figure, the single stage strategy was allowed to run for as many objective function evaluations as the two stage strategy. This advantage is also apparent from the hypervolume progressions of the single-stage and the two-stage approaches as depicted in Figure 6. Despite the additional complexity of the optimisation algorithm, we therefore adopt the two-stage approach.

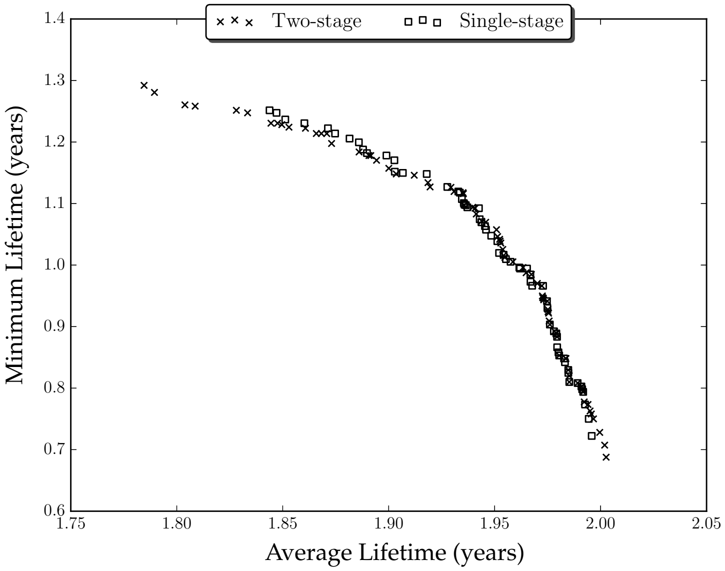


Figure 5: Comparison between single-stage optimisation and two-stage optimisation strategies. The single-stage approach performs as well as two-stage approach, except at the extreme minimum lifetime end.

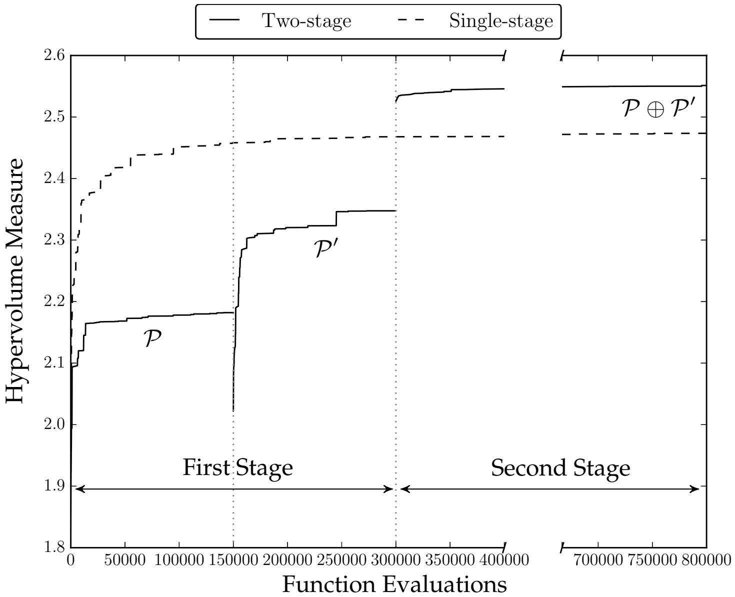


Figure 6: Comparison between the hypervolume progressions of two-stage (solid line) and single-stage (dashed line) approaches. In the first stage, optimising only in $\mathcal{P}$ (0 to 150,000) and $\mathcal{P}'$ (150,000 to 300,000) the hypervolume is close to convergence after about 150,000 function evaluations. At this point, switching to the combined $\mathcal{P}\oplus\mathcal{P}'$ provides an immediate improvement in hypervolume over the single-stage hypervolume. The single-stage technique is unable to match the two-stage with the same number of function evaluations, even when optimising using the combined path libraries $\mathcal{P} \oplus \mathcal{P}'$.

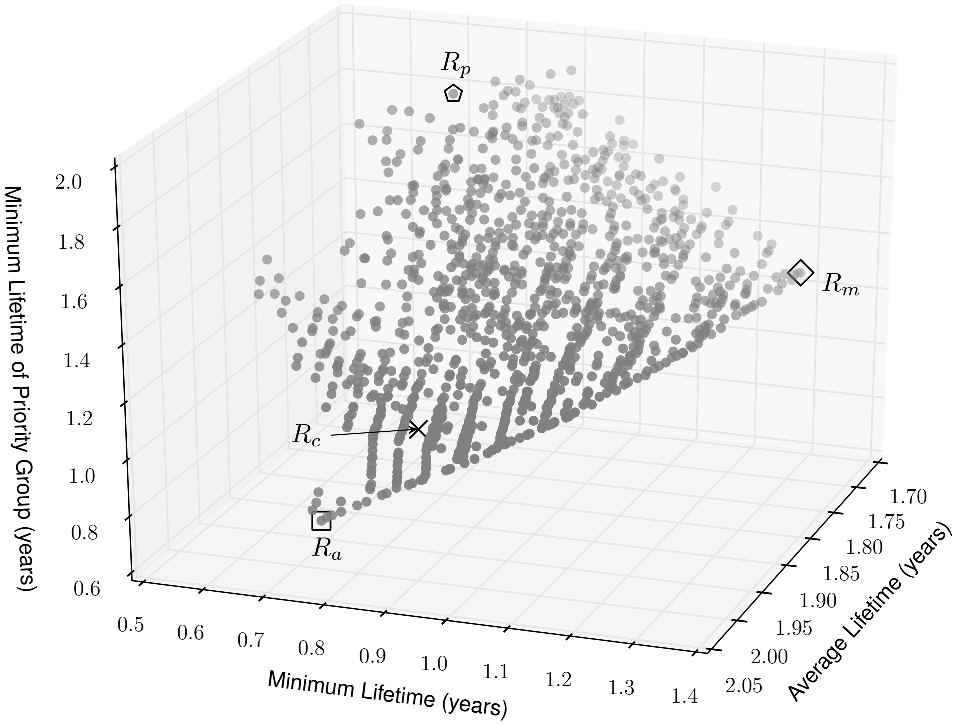


Figure 7: Pareto front approximation (grey shaded circles) with three objectives for the Victoria & Albert museum network. The solutions with best average lifetime, R_a, best overall minimum lifetime, R_m, and best minimum lifetime of the priority group, R_p, are indicated. The minimum energy solution R_c is dominated by all solutions in the Pareto front approximation.

6.2 Protecting a Group of Nodes

In practice, it is often desirable to give priority to one or more nodes so that their lifetimes are prolonged, because, for example, they are hard to access. To address this issue, we introduce an additional objective (7) with the aim of prolonging the minimum lifetime of the priority group of nodes, alongside aforementioned objectives.

We selected a priority group, $\mathcal{U} = \{v_8, v_{19}, v_{21}, v_{26}\}$ to include two heavily-loaded nodes (v_{19} and v_{21}) and two others. In Figure 7, we show the estimated trade-off surface in 3-dimensions between average lifetime, overall minimum lifetime and minimum lifetime of a priority group of nodes. Except for the additional objective, the hyper-parameters and general strategy for optimisation remain the same as before.

In Figure 8 we show the connectivity map as an adjacency matrix partitioned to place the priority group at the top left. The distinct nature of routing schemes corresponding to the extreme solution with respect to each of the objectives is depicted through the visualisation of the edge utilisations shown in Figure 9; here the grey scale indicates the frequency with which an individual radio link (edge) is used in a particular routing scheme.

To illustrate the underlying distinction in nature of the extremal solution for each

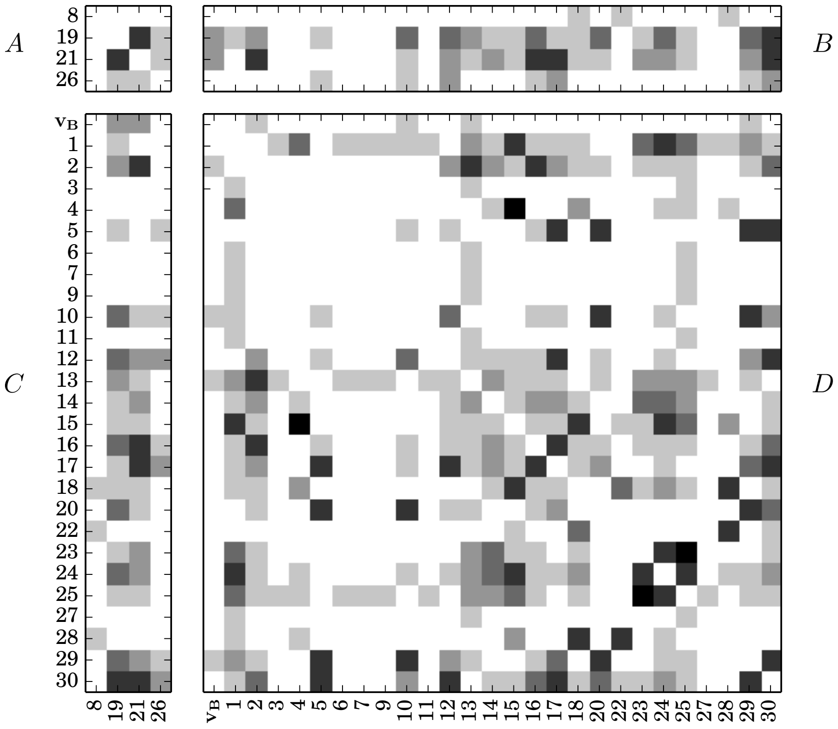


Figure 8: Adjacency matrix showing connectivity of nodes in the Victoria & Albert Museum network. The numbers on the left and bottom show the node indices; the base station is v_B. The grey scale indicates the energy consumption required for communication, with darker shades denoting less energy; white cells indicate there is no radio link between cells (infinite energy required for communication). The matrix is partitioned into submatrices showing connectivity between nodes in the priority group (A), nodes outside the priority group (D) and the inter-connectivity between priority and non-priority nodes ($B^\top = C$).

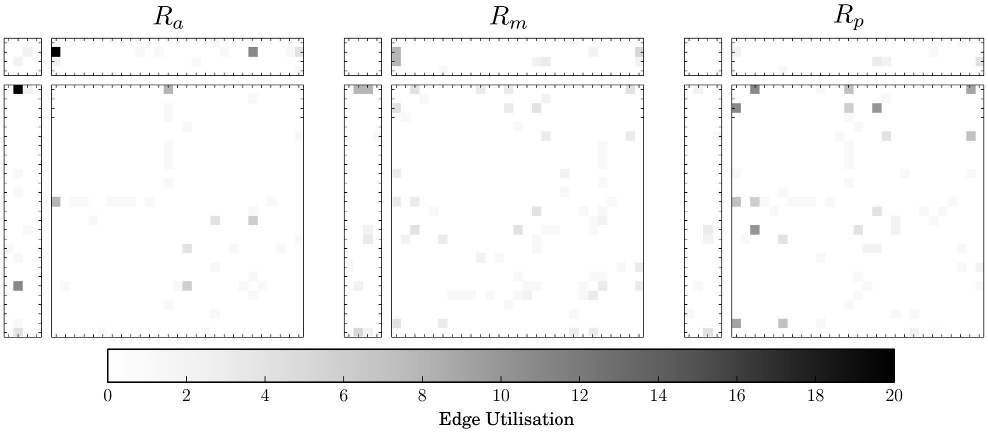


Figure 9: Edge utilisations for best average lifetime R_a, best overall minimum lifetime R_m and best minimum lifetime for the priority group R_p routing schemes. Higher edge utilisations are represented by darker greys.

objective, we consider the edge utilisations, i.e. the number of times a particular edge is used in a routing scheme, of three particular routing schemes: best average lifetime (R_a), best overall minimum lifetime (R_m) and best minimum lifetime of priority group (R_p) solutions. In Figure 9, we see that in R_a most paths utilise the link between node v_{19} and the base station v_B. As minimum energy routes are preferred, the average lifetime is high (1.99 years), but overall minimum lifetime and minimum lifetime of the priority group are low. The minimum lifetime is low as nodes on minimum energy paths relay most routes, and as a consequence exhaust their batteries quickly. The overall minimum lifetime and minimum lifetime of the priority group are the same, 0.73 years. This is because the most energy efficient routes heavily load v_{19}, which is in the priority group, so causing it to exhaust its battery first.

In contrast, R_m provides a more evenly distributed edge utilisation map, in particular v_{19} and v_{21} share the load between them. In this case, an increase of 78.1% in minimum lifetime when compared with R_a is achieved, at the expense of not using all energy efficient routes. Hence, the average lifetime is reduced by 12.5%. The minimum lifetime of the priority group is 1.3 years as v_{19} is the minimum lifetime node. We note that in this case the intra-connectivity between protected group nodes also diminishes.

The most dramatic difference is in R_p, where most paths avoid the priority nodes, improving the minimum lifetime for the priority group by 48.3% for v_{19} with respect to R_m. To achieve this increase in minimum lifetime for the priority group, high energy cost paths must be used for paths in the rest of the network resulting in a reduction in the minimum lifetime for the network as a whole; v_2 is the node whose battery is first exhausted in this case.

7 Better Approximation of Maximum Lifetime Routing

It is of considerable practical importance to maximise the minimum lifetime of any node in the network, because the network requires no maintenance until the battery of the shortest-lived node is exhausted. Chang and Tassiulas (2004) described how this can be modelled as an LP problem. The LP formulation relies on each node being able to use many paths to the base station, which in turn enables load balancing by using different routes for fractions of the network lifetime. However, the computational power and the storage requirements associated with switching between many routing schemes can be prohibitive for low-powered devices. Therefore, we consider a strategy in which only a small number of routing schemes (two or three) are available to the network, and all nodes change at pre-determined times to a new scheme. This re-routing can be accomplished at low energy cost from the base station. In this section we describe an evolutionary algorithm that optimises a small number of routing schemes together, and thus achieves a better approximation to the LP solution quality in comparison to using a single routing scheme throughout the lifetime of a network.

Intuitively, to see how using multiple routing schemes in a time-shared manner can be useful, suppose that under the initial routing scheme R, a node $v^\star$ has the minimum lifetime. Now suppose that the network is optimised a second time after it has been in operation for a while. The result of this second optimisation can be a routing scheme in which $v^\star$ is very lightly loaded, prolonging its life, while a different node, v', which was lightly loaded in the first epoch, now carries a heavier load. In this way the life of $v^\star$ is extended and it may be that $v^\star$ or v' or some other node is exhausted first, but in any case the time before any single node is exhausted is delayed.

7.1 Optimal Time Share

Inevitably the question now is how to best distribute the time share between multiple routing schemes. For a particular routing scheme, the current drain per reporting cycle is constant (the sum of a quiescent current together with transmission and reception costs to each of the nodes it communicates with). The decay in stored battery charge is therefore linear in time, although the decay rate depends on the particular routing scheme used. With this (good) assumption, the times for which each routing scheme should be used can be found by linear programming.

Consider a single routing scheme. If the initial charge at node v_i is q_{0i}, then after running the system with routing scheme R for time τ the charge remaining at the node is q_i:

$$q_i = q_{0i} + m_i \tau, \tag{21}$$

where, $m_i < 0$ is the rate of decay determined by the particular routing scheme in operation and given by $m_i = -(B_i + C_i)$ (c.f., equations (3) and (4)).

In a similar fashion, if there are D routing schemes, $R_1, \ldots, R_D$, in operation for times $\tau_1, \ldots, \tau_D$, the charge decay equation for v_i becomes:

$$q_i = q_{0i} + \sum_{d=1}^{D} m_{id} \tau_d. \tag{22}$$

Figure 10 illustrates the decay curve for v_i for multiple routing schemes.

Denoting the vector of time spans by $\boldsymbol{\tau} = (\tau_1, \tau_2, \ldots, \tau_D)^\top$, the vector of initial battery charges by $\mathbf{q}_0 = (q_{01}, q_{02}, \ldots, q_{0n})^\top$, the vector of remaining charges by $\mathbf{q} = (q_1, q_2, \ldots, q_n)^\top$ and the n by D matrix of decay rates by $\mathbf{M}$, the charge decay equations for all the nodes can be written as:

$$\mathbf{q} = \mathbf{q}_0 + \mathbf{M}\boldsymbol{\tau}. \tag{23}$$

The lifetime of the network is $\sum_{d=1}^{D} \tau_d = \mathbf{1}^\top \boldsymbol{\tau}$ subject to the constraint that the charge remaining at all nodes is positive. This allows us to formulate an LP problem for the timespans that will maximise the minimum lifetime of the network, given the D routing schemes as follows:

$$\max(\mathbf{1}^\top \boldsymbol{\tau}) \tag{24}$$

subject to:

$$\begin{aligned} \tau_d &\geq 0 \qquad d = 1, \ldots, D, && (25)\\ q_i &> 0 \qquad i = 1, \ldots, n. && (26) \end{aligned}$$

Solving this LP problem enables us to derive the optimal time span vector $\boldsymbol{\tau}^*$ for a given set of routing schemes.

The minimum lifetime node will exhaust its battery first. At this point the network would be re-routed for establishing connectivity for all other nodes as they have some charge left. However, discounting the re-routing, we can consider this residual charge at nodes for achieving better overall average lifetime. For such an arrangement, we can assume that the nodes with remaining charge will carry on with the last, i.e. Dth, routing scheme. Hence, the overall average lifetime will depend on the last routing

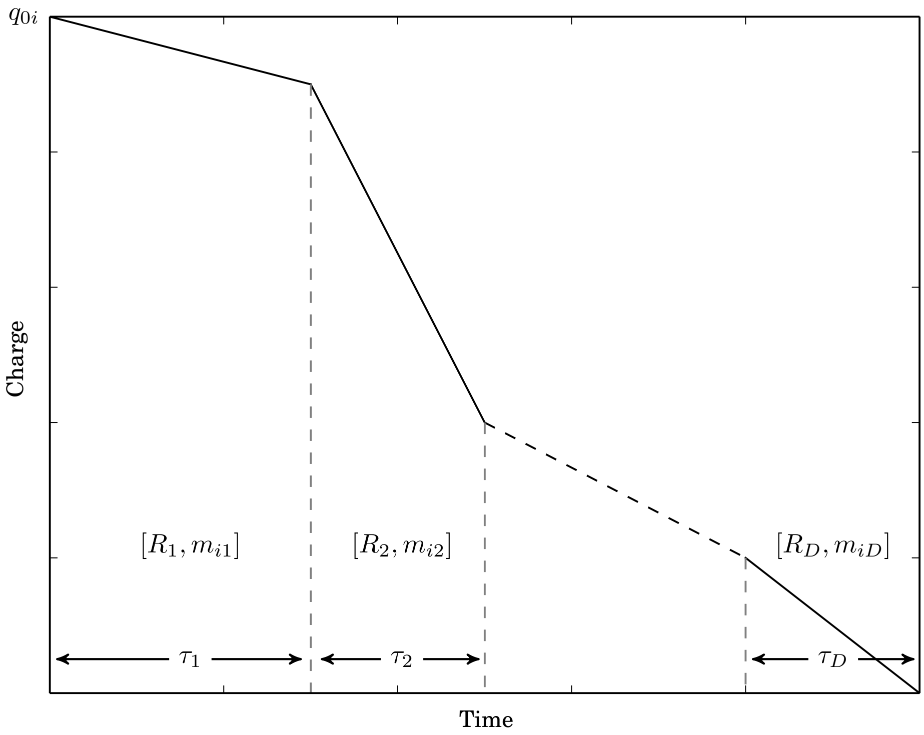


Figure 10: Battery charge decay for a node v_i with initial charge q_{0i} using multiple routing schemes. Routing scheme R_d is active for time span τ_d with a decay slope of m_{id}.

scheme in sequence. The time t_i remaining for node v_i after the minimum lifetime node is exhausted is given by

$$q_{0i} + \sum_{d=1}^{D} m_{id}\tau_d^* + m_{iD}t_i = 0. \tag{27}$$

The average lifetime of the network is then

$$\bar{\tau} = \mathbf{1}^\top \boldsymbol{\tau}^* + \frac{1}{n}\sum_{i=1}^{n} t_i. \tag{28}$$

Clearly, although changing the order in which the routing schemes are used does not affect the minimum lifetime, the average lifetime depends on which routing scheme is chosen as the last routing scheme, R_D. We therefore select R_D as the routing scheme that maximises the average lifetime.

Solving this LP problem and finding the routing scheme to use as R_D enables us to achieve the best minimum lifetime and average lifetime for given set of multiple routing schemes.

7.2 Evolving Multiple Routing Schemes

In order to locate good routing schemes we consider an evolutionary algorithm in which a solution $\mathbf{R}$ is constructed from D routing schemes: $\mathbf{R} = \{R_1, R_2, \ldots, R_D\}$. In this "$D$-RS" approach the evolutionary algorithm is used to optimise the routes comprising each of the constituent schemes, but optimal timespans of these evolved routes are found by solving the LP problem (24)-(26). This speeds up the search considerably compared with using an evolutionary algorithm or similar stochastic search to find good time spans as well as the good routing schemes. Note that it is possible for one or more timespans to be zero, so that the evolutionary algorithm can locate solutions in which only a single routing scheme is used.

Despite the pruning used to limit the size of the search space described in section 4, the use of several routing schemes leads to a further explosion in search space size because the number of paths available for each node is $2k$ in each of the D routing schemes (using k-shortest path pruning on G and G'). The number of possible solutions is thus approximately $(2k)^{nD}$, although some nodes may have fewer than $2k$ shortest paths available.

In order to combat this combinatorial explosion we adapt the basic evolutionary strategy for a single RS to perform a series of evolutionary searches in increasingly large search spaces as follows.

1. Evolutionary searches using path libraries $\mathcal{P}$ and $\mathcal{P}'$ are carried out separately but evolving single routing schemes rather than multiple schemes. (This can be achieved by replicating a single routing scheme D times and applying the same mutation and crossover operations to all of them.) As illustrated above, optimisation using $\mathcal{P}$ and $\mathcal{P}'$ separately helps to find good average lifetime solutions and good minimum lifetime solutions by restricting the size of the search space and guiding the search according to the pruning method used.

2. Still optimising using $\mathcal{P}$ and $\mathcal{P}'$ separately, we seek D-RS solutions. Here at each iteration mutation and crossover operations suggest new paths for nodes in one of the D constituent routing schemes, after which the optimal time spans, and thus the minimum and average lifetimes, are calculated by linear programming (24)-(26). As we illustrate later, solutions in which the routing schemes are the same throughout the lifetime of the network can produce better average lifetime solutions. Therefore, in half the iterations, a new solution is generated by replicating one of the component routing schemes D times.

3. Finally, the non-dominated subset of the union of the archives obtained in step 2 is used as the initial archive for a further optimisation using both $\mathcal{P}$ and $\mathcal{P}'$.

Recall that, as described in section 4.1, the path libraries were constructed from the k-shortest paths based on the composite cost. There the edge weights in (19) were calculated using the initial battery charges. However, the batteries become depleted by the network operation due to the routing scheme. We therefore, albeit infrequently (e.g. every 100,000 function evaluations), update the path libraries with k-shortest paths calculated on the basis of battery charge distributions from partially drained networks, and add any newly discovered paths to the path library. This allows the evolutionary mechanism to choose paths better suited to a network that has been in operation for some time.

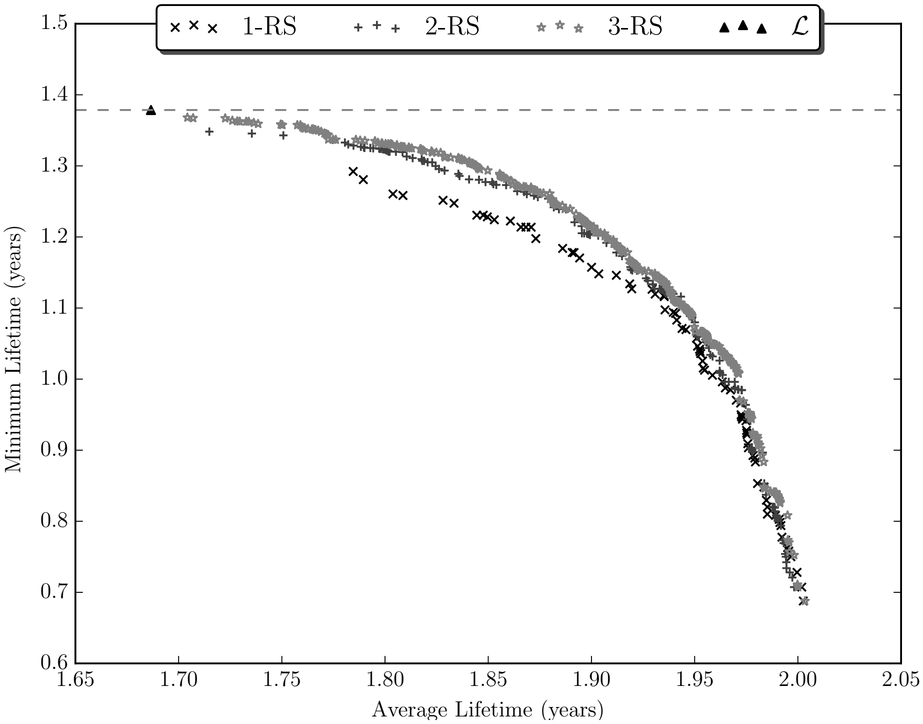


Figure 11: Performance comparison between optimisation with single routing scheme (1-RS, black crosses), two routing schemes (2-RS, dark grey pluses), and three routing schemes (3-RS, light grey stars). The performance of the maximum minimum lifetime solution with unlimited paths (∞-RS) is shown with a triangle. The horizontal dashed line through $\mathcal{L}$ shows the best possible minimum lifetime solution.

7.3 Performance Comparison

We compare the performance of optimised one, two and three routing schemes on the Victoria & Albert Museum network.

For both 2-RS and 3-RS strategies, we used path libraries generated with $k = 10$ shortest paths. The first stage of the optimisation (1-RS) was run for 150,000 iterations, while the second stage (multiple routing schemes using $\mathcal{P}$ or $\mathcal{P}'$ separately) were run for a further 800,000 iterations. Finally the third stage (multiple RS and combined path libraries) was run for an additional 1,000,000 iterations. These numbers of iterations yield well-converged fronts, and for practical purposes fewer iterations can obtain a good approximation.

In Figure 11, we show the estimated fronts for 1-RS, 2-RS, and 3-RS strategies. As may be expected, the estimated Pareto front for 2-RS dominates or equals the 1-RS scheme and it in turn is equalled or dominated by the 3-RS front. The 1-RS solution has a maximum minimum lifetime of 93.7% of the maximum lifetime routing solution, $\mathcal{L}$, which has the freedom to use unlimited routes. The 2-RS strategy achieves an improved minimum lifetime which is 97.8% of $\mathcal{L}$. The additional flexibility inherent in the 3-RS strategy allows a minimum lifetime that is 99.2% of $\mathcal{L}$ to be found.

In Figure 12 we present the summary attainment surfaces (Fonseca and Fleming,

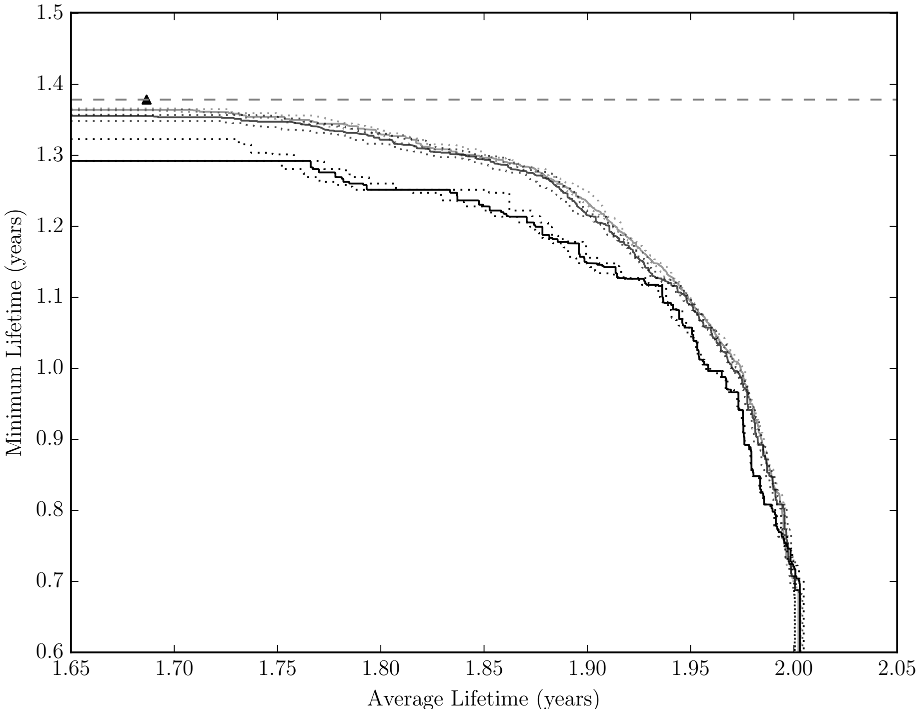
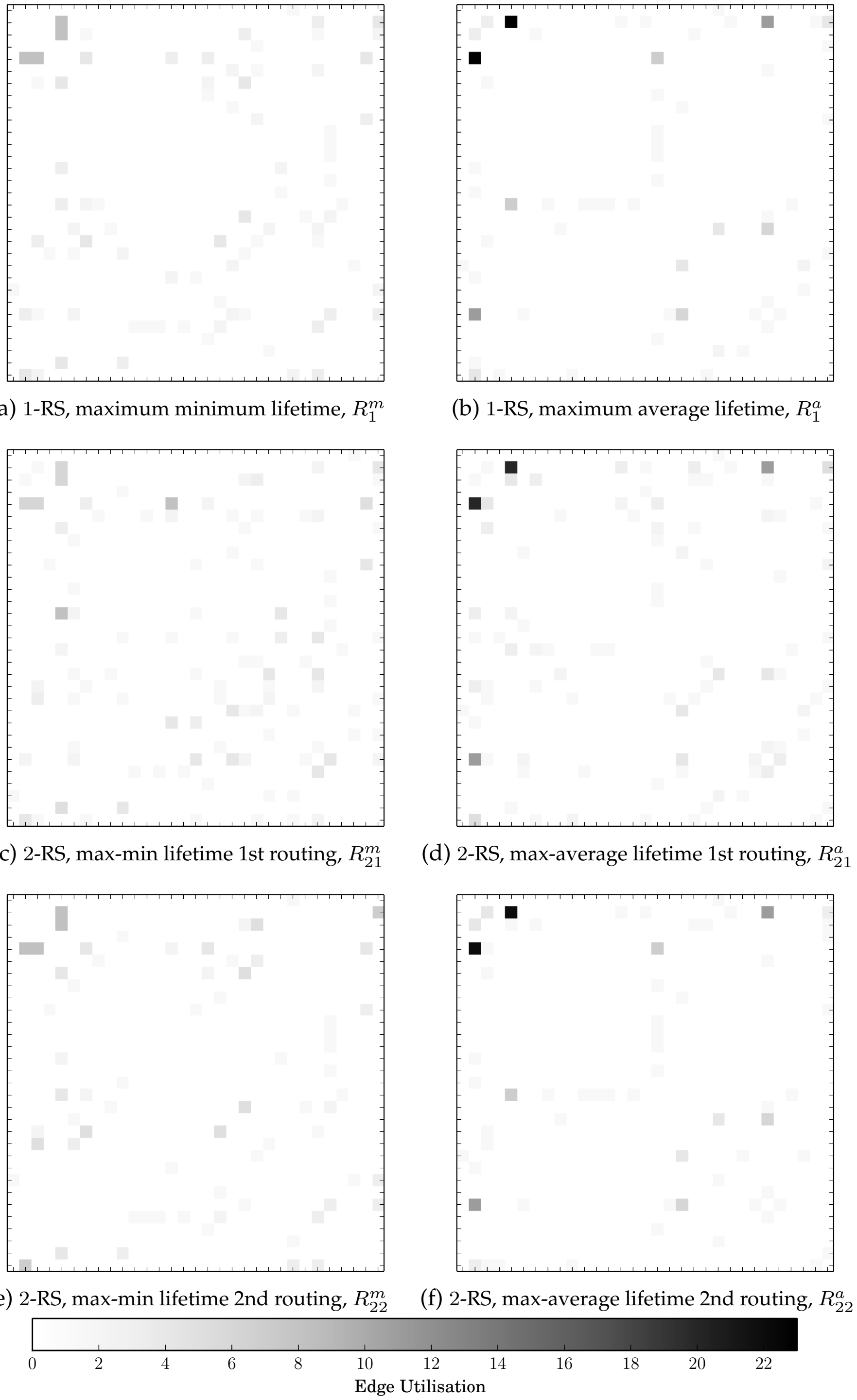


Figure 12: Performance comparison between 1-RS (black, bottom), 2-RS (dark grey, middle), and 3-RS (light grey, top) approaches in terms of summary attainment surfaces over 10 runs with solid lines representing 50% attainment surfaces, and dotted lines depicting 10% and 90% surfaces. The performance of the maximum minimum lifetime solution with unlimited paths is shown with a triangle. The horizontal dashed line through $\mathcal{L}$ shows the best possible minimum lifetime solution.

1996; Knowles, 2005) for 1-RS, 2-RS, and 3-RS strategies calculated over 10 runs. The narrow width between 10% and 90% attainment surfaces clearly indicates the desirable repeatability and convergence of this approach. It also shows that the approximation improves with each additional routing scheme; however, the rate of improvement diminishes as we increase the number of routing schemes.

Edge utilisations for best minimum lifetime and best average lifetime 1-RS and 2-RS solutions are shown in Figure 13. As noted previously, with a single routing scheme (top row of Figure 13) the best average lifetime solution heavily utilises a few edges, whereas the load is more evenly distributed for the maximum minimum lifetime solution. This pattern is also evident in the 2-RS solutions. The two component routing schemes, R^a_{21} and R^a_{22}, for the best average lifetime solution are both similar to the 1-RS solution. In fact, the 1-RS solution has a slightly better average lifetime than the 2-RS solution and in general we find that good average lifetime solutions tend to have a single RS.

Edge utilisations for the best minimum lifetime 2-RS component schemes R^m_{21} and R^m_{22} are shown in Figures 13c and 13e. These are active for 0.3 years and 1.05 years respectively, amounting to a total minimum lifetime of 1.35 years. However, individually



(a) 1-RS, maximum minimum lifetime, R_1^m

(b) 1-RS, maximum average lifetime, R_1^a

(c) 2-RS, max-min lifetime 1st routing, R_{21}^m

(d) 2-RS, max-average lifetime 1st routing, R_{21}^a

(e) 2-RS, max-min lifetime 2nd routing, R_{22}^m

(f) 2-RS, max-average lifetime 2nd routing, R_{22}^a

Figure 13: Multiple routing scheme edge utilisations. The top row shows edge utilisations for the best minimum lifetime R_1^m and best average lifetime R_1^a solutions using 1 routing scheme. Edge utilisations for the two components of the best minimum lifetime for a 2-RS solution shown in (c) and (e) and for the best average lifetime (d) and (f).

R^m_{21} and R^m_{22} can only provide minimum lifetimes of 0.88 and 1.29 years respectively, with the latter being the same as the best minimum lifetime solution from 1-RS (R^m_1 in Figure 13a). As the different distributions of edge utilisations in Figures 13c and 13e show, the additional lifetime is achieved by protecting some nodes while the first component is active, but then loading them more while the second component is active. Here these components are active for 22.2% and 77.8% of the network lifetime. The extra flexibility provided by a third component produces a small increase in achievable lifetimes, but the additional component is only used for a small fraction of time: the best minimum lifetime 3-RS scheme uses components for 6%, 18.7%, and 75.3% of the network lifetime.

Figure 11 and 12 also show that with each additional routing scheme in the solution, the approximation of the Pareto front improves. As well as improving the range of minimum lifetime solutions located, it also becomes denser. We attribute this to the additional flexibility to introduce paths that provide a finer grained trade-off between the objectives.

8 Conclusions

Using mesh network topologies in low power Wireless Sensor Networks poses a challenging problem of finding routes that best preserve the lifetime of individual nodes and the network as a whole. In this paper we have proposed a hybrid multi-objective evolutionary approach to approximate the optimum trade-off between the minimum lifetime for any nodes and the average lifetime of the whole, where these objectives are associated with infrequent maintenance and energy efficiency respectively. Although not discussed here, we remark that re-optimisation from a stored Pareto front approximation is an effective way of re-routing a network in case of node or link failure or if new nodes are introduced (Rahat et al., 2014).

Chang and Tassiulas (2004) have shown that the maximum lifetime routing for a network can be modelled as an LP problem that can be solved in polynomial time to obtain the proportions of messages that should be sent via different routes. However, this formulation requires that each node to be able to use many different paths to the base station. In low-power battery networks this is not feasible due the limited computational power and storage. Hence, we use a evolutionary approach to locate a set of routes for each node that can produce a close approximation to the many-paths LP solution (99.2% with 3-RS).

As a combinatorial problem, the search space size can increase drastically as the number of nodes increases. To reduce this potentially large space while retaining sensible solutions, we use k-shortest path pruning on the full connectivity graph and on a reduced graph. The k-shortest path pruning of the full graph limits the number of routes available to a node and thus limits the search space, while retaining solutions that are energy efficient. On the other hand, the graph reduction method retains paths which tend to yield good minimum lifetime routes. Central to the efficacy of our approach is optimisation in these limited search spaces, both for single and multiple routing schemes. In addition, solving exactly for the proportions of time that each component is active in a multiple routing scheme solution obviates the need for an additional, expensive stochastic search.

As we have shown, the flexibility of more than one routing scheme allows better solutions to be located. The nodes in the solutions here are constrained to change to another routing simultaneously. Current work focusses on relaxing this requirement, while still restricting the number of routes available to each node, in order to bring additional robustness against node or link failure to the network.

Acknowledgements

This work was supported by a Knowledge Transfer Partnership awarded to the University of Exeter and the IMC Group Ltd (KTP008748). We would like to thank Martin Hancock (Technical Director, The IMC Group Ltd.), and Neil Lundy (Engineering Manager, The IMC Group Ltd.) for their support and contributions. We would also like to thank Boris Pretzel (Chief Scientist, Victoria & Albert Museum), and Bhavesh Shah (Scientist, Victoria & Albert Museum), for facilitating the real network test. A patent application based on this work has been made (reference number: GB 1400353.7)

References

Bashyal, S. and Venayagamoorthy, G. K. (2007). Collaborative routing algorithm for wireless sensor network longevity. In *Intelligent Sensors, Sensor Networks and Information, ISSNIP'07.*, pages 515–520.

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press, New York, NY, USA.

Brander, A. W. and Sinclair, M. C. (1995). A comparative study of k-shortest path algorithms. In *Proc. 11th UK Performance Eng. Workshop*, pages 370–379.

Chang, J.-H. and Tassiulas, L. (2004). Maximum lifetime routing in wireless sensor networks. *Networking, IEEE/ACM Transactions on*, 12(4):609–619.

Coello Coello, C. A., Lamont, G. B., and Veldhuisen, D. A. V. (2001). *Evolutionary algorithms for solving multi-objective problems*. Springer.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2001). *Introduction to Algorithms*. MIT Press.

Das, I. and Dennis, J. (1997). A closer look at drawbacks of minimizing weighted sums of objectives for pareto set generation in multicriteria optimization problems. *Structural optimization*, 14(1):63–69.

Eppstein, D. (1998). Finding the k shortest paths. *SIAM Journal on Computing*, 28(2):652–673.

Fieldsend, J. E., Everson, R. M., and Singh, S. (2003). Using unconstrained elite archives for multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 7(3):305–323.

Fonseca, C. and Fleming, P. (1996). On the performance assessment and comparison of stochastic multiobjective optimizers. In Voigt, H.-M., Ebeling, W., Rechenberg, I., and Schwefel, H.-P., editors, *Parallel Problem Solving from Nature - PPSN IV*, volume 1141 of *Lecture Notes in Computer Science*, pages 584–593. Springer Berlin Heidelberg.

Förster, A. (2007). Machine learning techniques applied to wireless ad-hoc networks: Guide and survey. In *Intelligent Sensors, Sensor Networks and Information, ISSNIP'07*, pages 365–370.

Islam, O. and Hussain, S. (2006). An intelligent multi-hop routing for wireless sensor networks. In *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology Workshops*, pages 239–242.

Kamath, S. and Nasipuri, A. (2011). Integrated load balanced and energy aware routing in large scale wireless sensor networks. In *IEEE, 30th International Performance Computing and Communications Conference (IPCCC), 2011*, pages 1–8.

Knowles, J. (2005). A summary-attainment-surface plotting method for visualizing the performance of stochastic multiobjective optimizers. In *Intelligent Systems Design and Applications, 2005. ISDA '05. Proceedings. 5th International Conference on*, pages 552–557.

Kulkarni, R. V., Förster, A., and Venayagamoorthy, G. K. (2011). Computational intelligence in wireless sensor networks: A survey. *IEEE Comms Surveys & Tutorials*, 13(1):68–96.

Madan, R. and Lall, S. (2006). Distributed algorithms for maximum lifetime routing in wireless sensor networks. *Wireless Communications, IEEE Transactions on*, 5(8):2185–2193.

Rahat, A. A. M., Everson, R. M., and Fieldsend, J. E. (2014). Multi-objective routing optimisation for battery-powered wireless sensor mesh networks. In *Proceedings of the 2014 Conference on Genetic and Evolutionary Computation*, GECCO '14, pages 1175–1182, New York, NY, USA. ACM.

Rodoplu, V. and Meng, T. (1999). Minimum energy mobile wireless networks. *Selected Areas in Communications, IEEE Journal on*, 17(8):1333–1344.

Xue, F., Sanderson, A., and Graves, R. (2006). Multi-objective routing in wireless sensor networks with a differential evolution algorithm. In *Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control, ICNSC '06*, pages 880–885.

Yen, J. Y. (1971). Finding the K shortest loopless paths in a network. *Management Science*, 17:712–716.

Yetgin, H., Cheung, K. T. K., and Hanzo, L. (2012). Multi-objective routing optimization using evolutionary algorithms. In *Wireless Communications and Networking Conference (WCNC)*, pages 3030–3034.

Zang, I. (1980). A smoothing-out technique for min-max optimization. *Mathematical Programming*, 19(1):61–77.

Zitzler, E. (1999). *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*. PhD thesis, ETH Zurich, Switzerland.