Using Genetic Algorithms to Improve the Reliability of Dual Homed Wireless Critical Services

Eirik L. Følstad
Department of Telematics
Norwegian University of Science and Technology
Trondheim, Norway
Email: eirik.folstad@item.ntnu.no

Bjarne E. Helvik
Department of Telematics
Norwegian University of Science and Technology
Trondheim, Norway
Email: bjarne@item.ntnu.no

***Abstract*—The wireless access to any service in different contexts is nowadays taken for granted. However, the dependability requirements are different for various services and contexts. Critical services put high requirement on the service reliability, i.e., the probability of no service interruption should be close to one. Dual homing may be used to increase the service reliability in a multi technology, multi operator wireless environment, where the user's mobility necessitates access point selections and handovers. To allow the user to assess the risk of the service session, a prediction of the service reliability is necessary. This prediction must fulfil the need for the optimal sequence of access point selections and handovers with regard to service reliability and being computation efficient to accomplish the need for the real-time operation. We demonstrate how genetic algorithms (GA) may be used to predict and to improve the (near) optimal service reliability by fast and simple heuristics, far more computationally efficient than an Integer Linear Programming (ILP) optimization.**

***Index Terms*—Critical services, wireless networks, reliability, genetic algorithms, optimization**

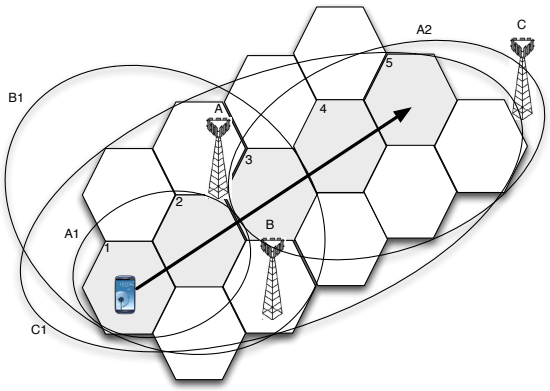

Fig. 1. Example network with three network operators (A, B and C) and their access points (A1 and A2, B1 and C1 respectively), with planned access coverage (ellipses). Along a projected route (arrow) the user moves through a number of virtual cells (hexagons).

I. Introduction

The evolution of wireless technologies, such as e.g., local area network (WLAN), High Speed Packet Access (HSPA) and Long Term Evolution (LTE) combined with the widespread use of smartphones [1] have made the wireless the preferred access service. Various contexts and services demand different dependability requirements.

For critical services, like emergency handling, health care services, energy control and surveillance/monitoring, the service reliability is crucial. In such contexts, the probability of no service interruption should be close to one. Dual homing protocols, such as mobile stream control transmission protocol [2] and Site multi homing by IPv6 Intermediation (SHIM6) [3], may be used to increase the service reliability where the diversity of the wireless accesses are utilized across different technologies and network operators.

Networks are evolving and network topologies change rapidly. As critical services may be started at any location there is a need at run-time to predict and to identify the access points and handovers along a projected route to allow the user to assess the risk before starting the critical service. A projected route is the physical movement of the user. Optional routes may for instance by found by means of navigation tools. For each of these there is a probability that the service may be completed without failures, i.e., no interrupts. A model for prediction of service reliability of a dual homed critical service is established [4].

The objective of this paper is to show that genetic algorithms (GA) may be used to effectively combine simple and fast heuristics to find an optimal or near optimal route. This approach closes the gap between the service reliability obtained by straightforward heuristics and the optimal obtained by ILP optimization, which is computationally too demanding in an operational system.

A trajectory is defined as series of access points used for each of the two radio connections for a dual homed critical service along a projected route. Most handover algorithms and dual homing protocols use local hop-by-hop based decisions and do not consider all handovers during a service session. The use of the optimal trajectory may be considered as a global based handover strategy optimized for service reliability.

The service reliability is measured by the metric $R(t_m) = P(T_{FF} > t_m)$ where T_{FF} is time to first failure and t_m is the mission time. Assume a network scenario as depicted in Fig.1. Planned access coverage, indicated by ellipses, is

provided by three operators named A, B, and C with their access points $A1$, $A2$, $B1$ and $C1$ respectively. The access points may provide coverage from different technologies. An arrow in Fig.1 represents the projected route. At the starting location, the objective is at run-time to derive the trajectory with (near) optimal service reliability.

The work in this paper is based on model for prediction of service reliability of a dual homed critical service in a multi technology, multi operator wireless environment, [4], briefly outlined in Section II. This prediction model defines the concept of virtual cells as limited geographical areas where the radio conditions are homogenous. Handovers are performed between the virtual cells and effectively limit the number of trajectories for a projected route. Virtual cells are indexed according to the order they are visited from 1 to m along the projected route. Access points covering virtual cell d is identified with the set b_d. Let a trajectory be identified by $S_{ij} = \{(i_1, j_1), \ldots, (i_d, j_d), \ldots, (i_m, j_m)\}$ where $i_d, j_d \in b_d$, $i_d \neq j_d$ and i and j are the two connections for the dual homed critical service. The user will traverse a number of virtual cells, shown as hexagons in Fig.1, along the projected route. The objective of this paper is to efficiently find the trajectory with the (near) optimal service reliability, i.e.,

$$\arg\max_{S_{ij}} R_{S_{ij}}(t_m) \tag{1}$$

In [5] the optimal trajectory was derived by ILP optimization using the model from [4]. It was shown that increased service reliability might be obtained with global handover strategy compared with a local hop-by-hop strategy when reliability is being optimized. Heuristics were proposed based on Dijkstra [6] and Bhandari [7] for computation efficiency on behalf of the optimality of the trajectory. We use GA combined with results of the heuristics in [5] to improve the (near) optimal trajectory while still being computational efficient compared with an ILP optimization.

The term genetic algorithm was coined by Holland [8]. An introduction to GA may be found in [9]. In [10] it is proposed how GA may be used for finding (near) optimal all-terminal network topology considering cost and reliability. A variable-length chromosome and population sizing are presented in [11] for shortest path routing. Both [10] and [11] pinpoint the challenges for efficient fitness function and control parameters. It is possible to identify the optimal GA control parameters for the desired optimization, but this may itself be a time demanding task [12]. Repair functions are described in [11] and [13] for chromosomes representing infeasible paths in networks. We propose a GA where chromosomes have identical length for a projected route, and the initial population is initialized with chromosomes from heuristics to allow fast convergence. The chromosome structure and the fitness function allow efficient crossover and mutation procedures without complex repair.

The rest of the paper is organized as follows. First in Section II we present the proposed model, while Section III describes the proposed GA and functionality. Several network scenarios are described and generated in IV with stochastic characteristics for verifying the performance of the GA in section V. Section VI concludes the paper.

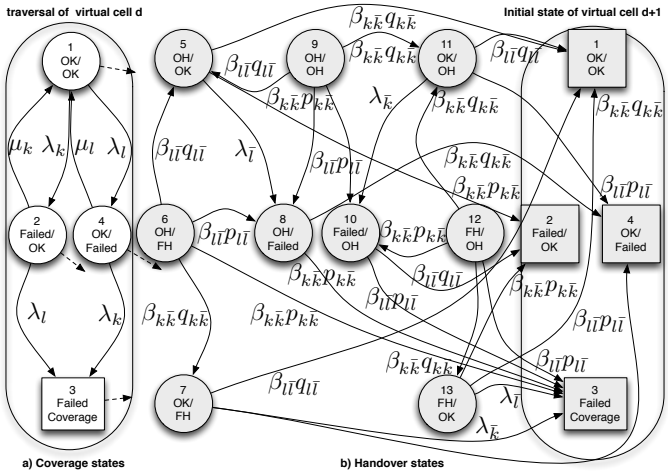

Fig. 2. State transition diagram for traversal of virtual cell d and handover to cell $d+1$. Absorbing handover states (grey shaded) represent the initial conditions for traversal of cell $d+1$. Transitions into handover states are modelled as instantaneous transition given by S_{ij}.

II. Dependability model of a trajectory

In the following we describe how the reliability of a trajectory of a dual homed critical service is predicted. The reliability will be used as the basis for fitness value for the chromosomes in the GA.

In [5] the reliability modelling of a trajectory is described as a phased mission. Each phase corresponds to handover from previous cell $d-1$ and traversal of virtual cell d, modelled as a continuous time discrete space Markov model as given in Fig.2. The names of the states are according to the state of the two connections of a trajectory S_{ij}. For instance, the state OK/OK indicates that both connections are working and the state FH/OH indicates that connection i has a failed radio connection and executing handover while connection j has a working radio connection and executing a handover. The dependability parameters and attributes are obtained from measurement reports for user equipment and signalling in the networks, as explained in [4]. Handover states $\Omega_H = \{5, 6, ..., 13\}$ are grey, while the white ones are coverage states $\Omega_C = \{1, 2, 3, 4\}$ since loss of radio connection is the main cause of failure within a virtual cell. Transitions into Ω_H are modelled as instantaneous transition given by S_{ij} and will be described later in this section. The change from phase d to $d+1$ takes place at time t_d and the sojourn time of phase d is T_d.

To keep the notation simple, we will in the rest of the paper use the index i_d for dependability parameters and attributes for an access point i_d in a virtual cell d, i.e., $\lambda_{id,d} \rightarrow \lambda_{id}$. In Fig.2 the indexes $k = i_d$, $\bar{k} = i_{d+1}$ and $l = j_d$, $\bar{l} = j_{d+1}$. A radio connection between a user equipment and an access point i_d covering virtual cell d has a constant radio coverage failure intensity λ_{i_d}. The radio coverage from different access points

fail independently. When a radio connection to access point i_d has been lost, the time to recover is n.e.d with mean $1/\mu_{i_d}$ in cell d. Handover time from i_d to i_{d+1} is n.e.d. with mean $1/\beta_{i_d i_{d+1}}$ and may fail with a probability of $p_{i_d i_{d+1}}$ where $q_{i_d i_{d+1}} = 1 - p_{i_d i_{d+1}}$. In case of dual handovers, the handovers fail independently.

Based on the model and results from [5], the reliability of a trajectory may be written as

$$\hat{R}_{S_{ij}}(t_{m_-}) = \hat{R}_1(T_{1_-}) \prod_{d=2}^{m} \hat{R}_d\left(T_{d_-} | T_{FF} > t_{d-1}\right) \quad (2)$$

where $\hat{R}_d(T_{d_-}|T_{FF} > t_{d-1})$ represents the reliability of phase d. The notations t_{d_-} and t_{d_+} are used to indicate instants immediately before and after the handover that takes place. Transition intensity matrices for Fig.2 may be organized as

$$\Lambda_C = \begin{bmatrix} \Lambda_{CC} & 0 \\ 0 & 0 \end{bmatrix} \text{ and } \Lambda_H = \begin{bmatrix} 0 & \Lambda_{HC} \\ 0 & \Lambda_{HH} \end{bmatrix} \quad (3)$$

where Λ_{CC} is the transition intensity matrix for Ω_C, Λ_{HH} is the transition intensity matrix for Ω_H and Λ_{HC} is the transition intensity matrix for intensities from Ω_H to Ω_C. As described in [4], the approximated normalized transient probability of working states $\hat{p}(T_{d_-}|T_{FF} > t_d) = \{\dot{p}_1, \dot{p}_2, 0, \dot{p}_4, 0, 0, 0, 0, 0, 0, 0\}^T$, where $\dot{p}_2 = \dot{p}_1 \lambda_{i_d}/\mu_{i_d}$ and $\dot{p}_4 = \dot{p}_1 \lambda_{j_d}/\mu_{j_d}$ and $\dot{p}_1 + \dot{p}_2 + \dot{p}_4 = 1$. The approximation holds as long as time spent in virtual cell T_d is more than the largest of $4/\mu_{i_d}$ and $4/\mu_{j_d}$.

By neglecting the handover time, the state just after handover is given as $p(T_{d_+}) = (\Pi_H)^3 p(T_{d_-}|T_{FF} > t_d, I_d)$ where Π_H is the transition probability matrix of Λ_H and $I_d = |i_d \cap i_{d+1}||||j_d \cap j_{d+1}|$ is the indicator function for the handovers. Only 3 operations are necessary since 3 is the largest path from an initial to an absorbing state in Ω_H. The instantaneous transitions from Ω_C to Ω_H are given by

$$\begin{aligned} p(T_{d_-}|T_{FF} > t_d, I_d = 11) &= \dot{p}(T_{d_-}|T_{FF} > t_d) \quad (4) \\ p(T_{d_-}|T_{FF} > t_d, I_d = 01) &= [0, \ldots, \dot{p}_1, 0, 0, \dot{p}_4, 0, 0, 0, 0, \dot{p}_2]^T \\ p(T_{d_-}|T_{FF} > t_d, I_d = 10) &= [0, \ldots, \dot{p}_4, 0, 0, \dot{p}_2, \dot{p}_1, 0, 0]^T \\ p(T_{d_-}|T_{FF} > t_d, I_d = 00) &= [0, \ldots, \dot{p}_4, 0, 0, \dot{p}_1, 0, 0, \dot{p}_2, 0]^T \end{aligned}$$

The traversal of cell d is given by $\Lambda_C p(t) = dp(t)/dt$ with the initial condition $p(0)$. For first phase $p(0) = \{1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}^T$ and for phase d is $p(0) = \{p_1(T_{d-1_+}), p_2(T_{d-1_+}), p_3(T_{d-1_+}), p_4(T_{d-1_+}), 0, \ldots, 0\}^T$. This yields the reliability of phase d

$$\hat{R}_d(T_{d_-}|T_{FF} > t_{d-1}) = 1 - p_3(T_{d_-}) \quad (5)$$

When the (near) optimal trajectory is found by using the approximation given by (5), the actual trajectory reliability , $R_{S_{ij}}(t_{m_-})$, may be calculated by using $\hat{p}(T_{d_-}|T_{FF} > t_d) = \{p_1(T_{d_-}), p_2(T_{d_-}), 0, p_4(T_{d_-}), 0, 0, 0, 0, 0, 0, 0\}^T$.

In the following we will describe how $\hat{R}_{S_{ij}}(t_{m_-})$ is used by GA.

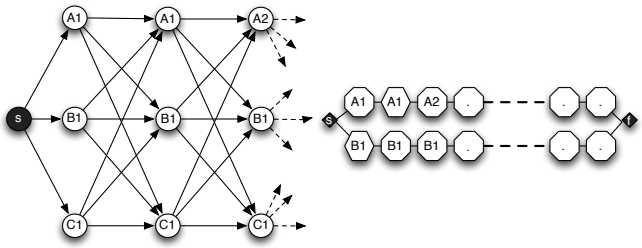

Fig. 3. A graph model of the example network in Fig.3 with sketch indicating all possible trajectories and a chromosome representing one specific trajectory.

III. Description of GA

Each of the main procedures in GA will now be described along with the use of the heuristics in [5] to improve the (near) optimal trajectory for a dual homed critical services. First the chromosome structure and the related fitness value are introduced.

A. Chromosome and fitness

It is seen that we may let the trajectory S_{ij} represents a chromosome, where (i_d, j_d) being the d'th gene-pair. The example network in Fig.1 is represented by a graph model as shown in Fig.3 that indicates all possible trajectories. One specific trajectory is shown as a chromosome in the figure where its third gene-pair is $(A2, B1)$.

A chromosome S_{ij} is given a fitness value according to the function

$$f_{S_{ij}} = \frac{1}{1 - \hat{R}_{S_{ij}}(t_{m_-})} \quad (6)$$

The chromosome representing the trajectory with the highest reliability is given by $\arg\max_{\forall S_{ij}} \hat{f}_{S_{ij}}$.

For making the GA computationally efficient the approximation of the reliability derived in (2) is used. The exact service reliability could be used for the chromosome fitness, but this would be far more computation demanding.

B. Initialization

The initialization procedure introduces a population, called generation 1 as G_1, with $|G_1| = n$ feasible chromosomes, where each chromosome S_{ij} fulfils the constraints

$$|S_{ij}| = m, \forall (i, j) \quad (7)$$

$$\forall (i_d, j_d) \in S_{ij}, \text{where } i_d, j_d \in b_d, \forall d = 1, \ldots, m \quad (8)$$

$$\forall (i_d, j_d) \in S_{ij}, \text{where } i_d \neq j_d, \forall d = 1, \ldots, m \quad (9)$$

The size of the population, n, is be kept constant during the evolution of the population.

The goal with the GA is to close the gap between the service reliability of the trajectories obtained with the heuristics and ILP optimization in [5] and still be computation efficient. The initial population may therefore be initialized with chromosomes representing the trajectories obtained with heuristics (further explained in section V). A random initialized population may also be generated to benchmark the use of the heuristics initialization.

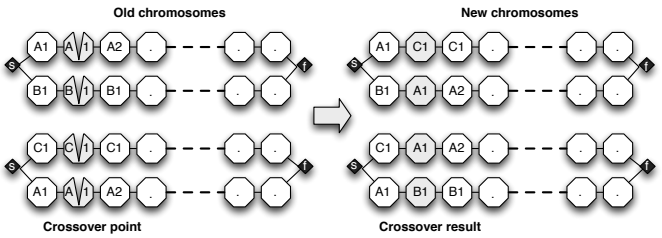

Fig. 4. Example of crossover between two chromosomes of Fig.3.

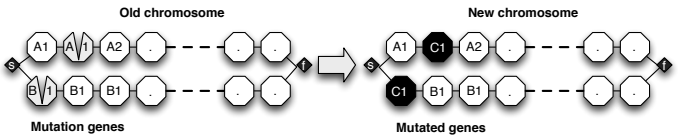

Fig. 5. Example of mutation at two genes of a chromosome of Fig.3.

C. Selection

The selection is either a roulette or a tournament process. Both selection processes allow some chromosomes to be selected several times and others to not be selected at all.

For the roulette selection the probability $p_{S_{ij}}$ for choosing the chromosome S_{ij} from the population G_c is proportional to its relative fitness, i.e.,

$$p_{S_{ij}} = \frac{f_{S_{ij}}}{\sum_{\forall S_{kl} \in G_c} f_{S_{kl}}} \tag{10}$$

From generation G_c n chromosomes are randomly selected according to their relative fitness and constitutes the intermediate population $\hat{G}_c$.

For the tournament selection process, n pairs of chromosomes are compared and the chromosome with the highest fitness from each pair value is selected. A chromosome is appointed with probability $1/n$ for a tournament. The winners of all tournaments constitute the intermediate population $\hat{G}_c$.

D. Crossover

After the selection process described in Subsection III-C the intermediate population $\hat{G}_c$ undergoes a crossover procedure. From the randomly ordered set of chromosomes consecutive pairs of chromosomes are selected for crossover with an i.i. probability p_c. Say that the chromosomes S_{ij} and S_{kl} are selected for crossover. The crossover point $x \sim$ uniform$[1, m]$ and the two new chromosomes are obtained as $S_{\hat{i}\hat{j}} = \{(i_1, j_1), \ldots, (k_x, l_x), \ldots, (k_m, l_m)\}$ and $S_{\hat{k}\hat{l}} = \{(k_1, l_1), \ldots, (i_x, j_x), \ldots, (i_m, j_m)\}$.

In Fig.4 the crossover operation between a pair of chromosomes from Fig.3 is depicted. At the left hand side of the figure the marked chromosomes and the gene-pair position 2 for crossover are identified. The halves defined by the crossover point are interchanged between the chromosomes, as shown at the right hand side of the figure. Note that since a handover between $i_d \in b_d$ and any $i_{d+1} \in b_{d+1}$ is possible and since $|S_{ij}| = |S_{kl}|$ there is no need for a repair after a crossover.

E. Mutation

After the crossover procedure the intermediate population $\hat{G}_c$ undergoes a mutation procedure. Unlike the crossover, the mutation procedure allows for mutation of multiple genes in a chromosome. Each gene of a chromosome is mutated with an i.i. probability p_m. Define an indicator function $I(ij)$ for the chromosome S_{ij} where $I(i_d) = 1$ if mutation is introduced for gene i_d of connection i. Likewise is defined for $I(j_d)$. Let's say a mutation is introduced only in gene i_d of chromosome S_{ij}. If $|b_d| > 2$ the mutated chromosome, $S_{ij}^{I(ij)}$, becomes $S_{ij}^{I(ij)} = \{(i_1, j_1), \ldots, (k_d, j_d), \ldots, (i_m, j_m)\}$ where $k_d \sim$ uniform $[b_d \setminus \{i_d, j_d\}]$. For the cases where $|b_d| = 2$ then $S_{ij}^{I(ij)} = \{(i_1, j_1), \ldots, (j_d, i_d), \ldots, (i_m, j_m)\}$. This may be seen as a simple repair for not violating constraint (9).

A mutation of a chromosome S_{ij} from Fig.3 is shown in Fig.5. Here the mutations are marked at the lower half of gene-pair 1 and at the upper half of gene-pair 2, i.e., $I(j_1) = 1$ and $I(i_2) = 1$, as shown at the left hand side of the figure. At the right hand side of the figure the mutated genes are shaded black for the mutated chromosome $S_{ij}^{I(ij)}$. For instance, to mutate gene i_2 the possible access points are $b_2 \setminus \{A1, B1\}$ that reduces to $\{C1\}$ since $b_2 = \{A1, B1, C1\}$.

F. Elitism

In our implementation of the GA we can use none, simple or global elitism. In all cases the best fitted chromosome found in any generation is stored.

Without elitism the new population is equal the intermediate population after the crossover and mutation procedures, where $G_{c+1} = \hat{G}_c$.

When simple elitism is used the new population is the best fitted chromosomes from previous population and $n - 1$ best chromosomes from intermediate population, i.e., $G_{c+1} = (S_{ij} \cup (\hat{G}_c \setminus S_{xy})$, where $S_{ij} = \arg\max f_{S_{kl}}, \forall S_{kl} \in G_c$ and $S_{xy} = \arg\min f_{S_{kl}}, \forall S_{kl} \in \hat{G}_c$.

With global elitism the new population is the n best fitted chromosomes from the previous and the intermediate population.

IV. Reference cases for comparisons

To compare the reliability of the trajectory found by GA with the optimal trajectory found by ILP optimization, we define a number of scenario-classes adapted from [5]. The scenario-classes are basis for scenario-instances that represent a graph model of a network, similar as shown in Fig.3.

In [5] it is described six scenario-classes, each with a defined number of virtual cells and network operators providing radio coverage with maximum one access point each per virtual cell. A network operator provides radio coverage for successive virtual cells along the projected route where one access point may cover several successive virtual cells. In cases where a network operator provides coverage for only a part of the projected route, the first virtual cell is randomly selected. The scenario-class definitions are given in Table I.

TABLE I
SCENARIO-CLASSES AND OPERATOR ACCESS COVERAGE

		Characteristics							
		Coverage Operator A		Coverage Operator B		Coverage Operator C		Coverage Operator D	
Class	Cells	cells per AP	total cells	cells per AP	total cells	cells per AP	total cells	cells per AP	total cells
1	5	U[2,3]	5	2	2	5	5		
2	10	as class 1	10	as class 1	4	as class 1	10		
3	15	as class 1	15	as class 1	6	as class 1	15		
4	5	U[2,3]	5	2	2	5	5	U[1,2]	2
5	10	as class 4	10	as class 4	4	as class 4	10	as class 4	4
6	15	as class 4	15	as class 4	6	as class 4	15	as class 4	6

TABLE II
DEPENDABILITY PARAMETERS FOR SCENARIO-INSTANCES

	Parameters	Values (for λ, μ and β the unit is s^{-1})			
		$i_{d+1} \in b^A_{d+1}$	$i_{d+1} \in b^B_{d+1}$	$i_{d+1} \in b^C_{d+1}$	$i_{d+1} \in b^D_{d+1}$
	$\lambda_{i_{d+1}}$	$U[1/998, 2/998]$	$U[2/998, 5/998]$	$U[1/998, 3/998]$	$U[1/998, 2/998]$
	$\mu_{i_{d+1}}$	$U[1/4, 1/2]$	$U[1/4, 1/2]$	$U[1/4, 1/2]$	$U[1/4, 1/2]$
$i_d \in b^A_d$	$p_{i_d i_{d+1}}$	$U[0.01, 0.02]$	$U[0.01, 0.04]$	$U[0.01, 0.03]$	$U[0.01, 0.04]$
	$\beta_{i_d i_{d+1}}$	$U[4, 8]$	$U[2, 4]$	$U[4, 8]$	$U[2, 4]$
$i_d \in b^B_d$	$p_{i_d i_{d+1}}$	$U[0.01, 0.04]$	$U[0.01, 0.03]$	$U[0.01, 0.04]$	$U[0.01, 0.05]$
	$\beta_{i_d i_{d+1}}$	$U[3, 6]$	$U[4, 8]$	$U[3, 6]$	$U[3, 6]$
$i_d \in b^C_d$	$p_{i_d i_{d+1}}$	$U[0.01, 0.03]$	$U[0.01, 0.04]$	$U[0.01, 0.04]$	$U[0.01, 0.04]$
	$\beta_{i_d i_{d+1}}$	$U[2, 4]$	$U[4, 8]$	$U[4, 8]$	$U[2, 4]$
$i_d \in b^D_d$	$p_{i_d i_{d+1}}$	$U[0.01, 0.04]$	$U[0.01, 0.02]$	$U[0.01, 0.02]$	$U[0.01, 0.03]$
	$\beta_{i_d i_{d+1}}$	$U[3, 6]$	$U[3, 6]$	$U[2, 4]$	$U[4, 8]$

Note that only network operators A and C provide coverage for all virtual cells. From each of the scenario-classes 100 scenario-instances were created with dependability parameters as defined in Table II where all parameters are i.i.d. uniform distributions and where sojourn time T_d in each virtual cell is i.i.d. $\sim$ uniform$[20, 30]$ seconds. The access points covering a virtual cell d is given by $b_d = b^A_d \bigcup b^B_d \bigcup b^C_d \bigcup b^D_d$ where b^A_d, b^B_d, b^C_d and b^D_d are the set of access points for the different network operators covering virtual cell d.

For 80 of total 600 scenario-instances, none of the seven heuristics in [5] found the optimal trajectory as identified by the ILP optimization. These 80 scenario-instances are the reference cases used for comparing the performance of GA with the heuristics and ILP. The reference cases are numbered from 1 to 80, where reference case numbered 1 belongs to scenario-class 1, references $2, \ldots, 10$ to class 3, $11, \ldots, 14$ to class 4, $15, \ldots, 36$ to class 5 and $37, \ldots, 80$ to class 6.

V. RESULTS

In the following, the reliability of the trajectories for different scenario-instances found by the GAs are compared with the optimal trajectories as identified by the ILP optimization. Details of the ILP optimization are found in [5].

A total of 22 different GA control parameter sets were defined and are summarized in Table III. The sets are named according to values of the control parameters, for example R-H-95-03-G defines roulette (R) selection, heuristics (H) initialization, $p_c = 0.95$, $p_m = 0.03$ and global (G) elitism. The heuristics initialization refers to an initial population with the seven chromosomes representing the results from the seven heuristics in [5] along with 93 randomly created chromosomes. All parameters sets have a population size of $n = 100$.

TABLE III
OVERVIEW OF TOTAL 22 DIFFERENT GA CONTROL PARAMETER SETS

Selection	Initialization	Crossover, p_c	Mutation, p_m	Elitism
{Roulette,Tournament}	Heuristics	0.95	0.01	{None, Simple, Global}
{Roulette,Tournament}	Heuristics	0.70	0.01	{None, Simple, Global}
{Roulette,Tournament}	Random	0.95	0.01	{None, Global}
{Roulette,Tournament}	Random	0.70	0.001	{None, Global}
{Roulette,Tournament}	Heuristics	0.95	0.03	Global

For each of the reference cases as defined in Section IV 100 replications of the GA are run with exit criteria of maximum 100 generations or when the difference between the max and average fitness of a generation is less than 10^{-9}. The GA was implemented in Mathematica 8 [14] running on a PowerEdge M610 blade with 2.67GHz quad-core CPU with 24GB memory and a 64 bits Linux kernel.

A. Heuristics (H) vs. random (R) initialization

The objective is to investigate whether the initialization based on the heuristics, allows the GA to find the optimal trajectory as identified by an ILP optimization or a (near) optimal trajectory better than those found by the heuristics. A heuristics initialized GA provides at least as reliable trajectory as the best of the heuristics since the best fitted chromosome are always stored during the evolution of the population.

To visualise the reliability of the trajectories obtained by the GA with the parameter set R-H-95-03-G for the different reference cases, a boxplot is presented in Fig.6. The boxplot includes the 25% and 75% quantiles with the median and outliers of the difference between reliability of the trajectories found by the each replication of R-H-95-03-G and the optimal trajectory found by the ILP optimization, i.e., $R_{S_{ij}}(t_{m\_})_{ILP} - R_{S_{kl}}(t_{m\_})_{GA}$. The reference cases within each scenario-class are sorted according to the medians of the differences. In the figure the differences between the trajectory reliability obtained by ILP optimization and the best of the heuristics are depicted as a dashed line for each reference case. The reliability of the optimal trajectory identified by the ILP optimization is represented with the solid line in the figure. As may be seen in Fig. 6, the parameter set R-H-95-03-G finds the optimal trajectories except for the reference case 73 and for the reference case other than 78, 79 and 80, the 75% quantiles of the differences are less than $1 \cdot 10^{-4}$. As indicated in Fig. 6, the differences of the reliability of the trajectories found by ILP optimization and the set R-H-95-03-G are not directly related to the reliability of the optimal trajectory identified by the ILP optimization.

The parameter set R-H-95-03-G obtained the overall best results, determined by medians and standard deviations of the reliability of the trajectories for the reference cases, compared with other parameter sets using heuristics initialization, as classified in Table III. It may be noted that a) GA with parameter sets using heuristics initialization without elitism obtained significantly worse results than the R-H-95-03-G while b) the sets T-H-95-03-G, R-H-95-01-G and T-H-95-01-G

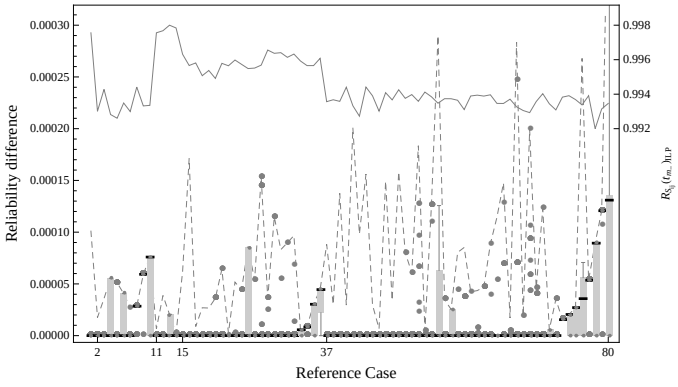

Fig. 6. Boxplot of $R_{S_{ij}}(t_{m\_})_{ILP} - R_{S_{kl}}(t_{m\_})_{GA}$ with median and outliers for parameter set R-H-95-03-G. The dashed line represents the differences between the ILP optimization and the best of the heuristics. The solid line represents the reliability of the optimal trajectory identified by the ILP optimization. Observe that the optimal trajectories except for the reference case 73 are found. In addition, except for the reference case $78, 79, 80$, the 75% quantiles of the differences are less than 10^{-4}.

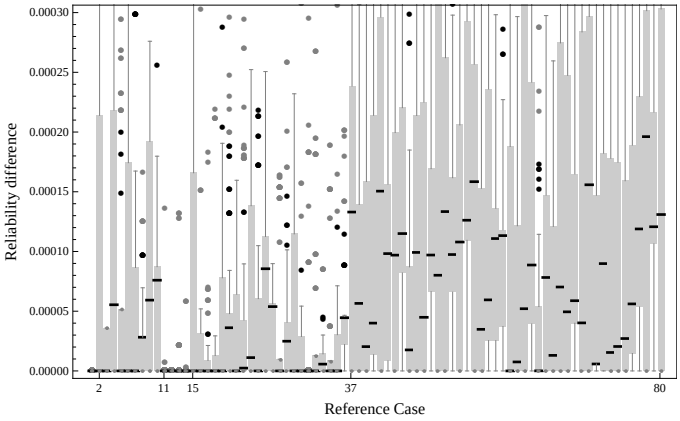

Fig. 7. Boxplot of $R_{S_{ij}}(t_{m\_})_{ILP} - R_{S_{kl}}(t_{m\_})_{GA}$ with median and outliers for parameter set R-R-95-01-G. Observe that the parameter set R-R-95-01-G only rarely finds the to the optimal solution as identified by ILP optimization and significantly less frequent than the set R-H-095-03-G.

obtained slightly reduced results, but specific reduced results for reference case 80.

To investigate whether the global elitism combined with heuristics initialization cause the GA to converge to local optimums, the initialization based on heuristics was omitted. The resulting boxplot is shown in Fig. 7 for the set R-R-95-01-G. As may be seen, the overall results, determined by medians and quantiles for the trajectories reliabilities obtained, get significantly worse than the set R-H-95-03-G. Observe that the parameter set R-R-95-01-G only rarely finds the optimal solution as identified by the ILP optimization and significantly less frequent than the set R-H-095-03-G. For the parameter sets with random initialization without elitism the results were even significantly worse than the set R-R-95-01-G. Using simple elitism, the parameter sets T-H-95-01-S and R-H-95-01-S, improved the means and standard deviation of the trajectories reliabilities for the reference cases $78, 79$ and 80, but overall they found less optimal trajectories for the other reference cases.

With the GA parameters sets and the reference cases studied these results indicate that a GA with heuristics initialization combined with global elitism is a good candidate for finding (near) optimal trajectories.

B. Computation effort

In Section V-A the reliability of a trajectory found by GA was compared with the optimal trajectory found by the ILP optimization. We now compared the computation effort of the GA compared with an ILP optimization [5].

As indicated in Fig.6, a number of replications of the GA may be needed to ensure that the (near) optimal trajectory is found for each of the reference cases. The difference of the reliability of the trajectories found by ILP optimization as obtained by the percentage of all replications of the set R-H-95-03-G is depicted in Fig. 8. Similar differences are indicated for the best of the seven heuristics for each reference case. Likewise, the Fig. 8 depicts the differences for scenario-class 6 only for both the set R-H-95-03-G and best of the seven heuristics. As may be seen in the figure, approximately 80% of all R-H-95-03-G replications find the same trajectory as the ILP optimization for all scenario-classes and scenario-class 6 only. On the opposite side, approximately 80% of the heuristics have differences of more than $1 \cdot 10^{-4}$.

The exact time consumption is not of the main interest, but how the computation effort changes with increasing complexity of the scenario-instances. For having an equal HW platform for the computation effort comparisons, the ILP optimizations in [5] were recomputed with the same HW as the GA implementation and the heuristics. The ILP optimizations were solved with the modelling language AMPL, version 20131213, with the commercial solver Gurobi, version 5.6.0.

The time complexity of the GA with parameter set R-H-95-03-G is analysed as follows. The heuristics initialization is dominated by the Bhandari algorithm which takes $O(b+m)^2$ time [5], where $b = |\bigcup_{d=1}^{m} b_d|$. For the GA the roulette wheel selection takes $O(n \log n)$ time, crossover operation $O(n)$ time, mutation operation $O(nm)$ time and elitism $O(n \log n)$ time. Combined together, the GA with parameter set R-H-95-03-G takes $O(b+m)^2 + O(nm)$ time, where $O(nm)$ is the time needed for each replication.

The mean computation effort for one replication of GA with the parameter sets R-H-95-03-G and R-R-95-01-G is depicted in Fig.9 for each of the reference cases. The computation time for ILP optimization and total computation time for the heuristics are also indicated. For a GA with heuristics initialization, the computation time for the heuristics is needed regardless of the number of replications. The mean computation time is comparable when using R-H-95-03-G and R-R-95-01-G. As may be observed in the figure, the computation time correlates with the complexity of the scenario-classes.

As indicated in Fig.9 the ILP optimization computation times for the scenario-class 6 reference cases have significant differences, ranging from approximately 150 seconds to more

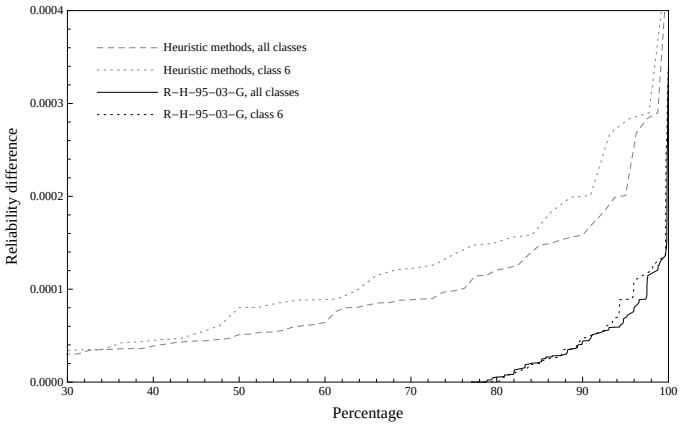

Fig. 8. The difference of the reliability of the trajectories found by ILP optimization as obtained by the percentage of all replications and the set R-H-95-03-G and the best of the heuristics. Observe that approximately 80% of all R-H-95-03-G replications find the same trajectory as the ILP optimization, while 80% of the heuristics have differences of more than $1 \cdot 10^{-4}$.

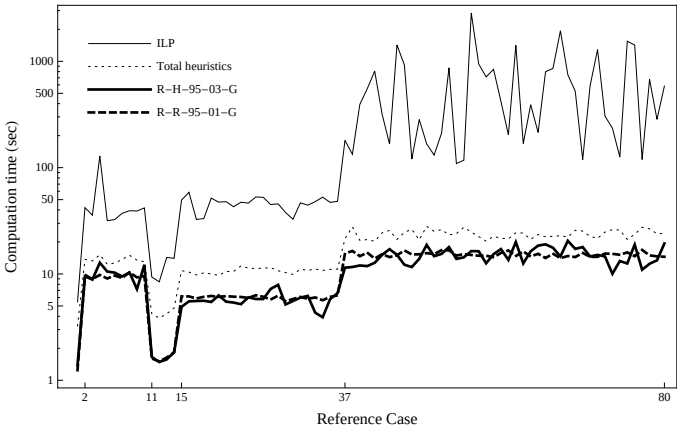

Fig. 9. The mean computation effort for GA with parameter sets R-H-95-03-G and R-R-95-01-G. The computation effort for ILP optimization and total time for the heuristics are also shown. Observe that the heuristics and the GA are less sensitive to the scenario-class complexity than the ILP optimization.

than 3000 seconds. The ILP optimization computation time is approximately five times higher for a class 5 instance than a class 4 instance, and on average approximately ten times higher for a class 6 instance than a class 5 instance. The GAs with the given parameter sets are not that sensitive to the complexity of the reference cases as the ILP optimization. For instance, the set R-H-95-03-G computation effort is approximately 2.5 times higher for a class 6 instance than a class 5 instance. When using random initialization with no elitism the R-R-95-01-N increased the computation time with approximately five times compared with the R-H-95-03-G.

Given the mean computation time for one replication of R-H-95-03-G combined with the percentage of replications obtaining the optimal trajectory, this indicates that a better (near) optimal trajectory may be found by the GA than the heuristics alone and with less computation effort than ILP optimization. For instance, using the results depicted in Fig.8 the number of replications needed by the R-H-95-03-G to find the optimal solution may be estimated as independent Bernoulli trials with success probability of 80%.

VI. Conclusion

For critical service the probability of no service interruption should be close to one. The use of the proposed GA is computationally efficient. In most cases it finds optimal or close to optimal trajectories when seeded by simple heuristics. This approach is not computationally demanding, i.e., very fast, and hence, it is extremely useful for practical implementation. A few cases are identified, where there is a notable gap between what is found by this method and the optimal trajectories. These are studied in depth, and we have not been able to get a significant improvement by using the full potential of GA, through large initial populations, many replications and parameter tuning.

The proposed method is a very good trade off between computationally efficiency and optimality of the found trajectory. It is sufficiently accurate and efficient for use in provision of dual homed wireless critical services with non stationary user equipment.

References

[1] H. Blodget, P. Gobry, and A. Cocotas, "The future of mobile," Business Insider, 2012.

[2] R. Stewart, Q. Xie, K. Morneault, C. Sharp, H. Schwarzbauer, T. Taylor, I. Rytina, M. Kalla, L. Zhang, and V. Paxson, "RFC2960: Stream control transmission protocol," 2000.

[3] M. B. E. Nordmark, "Shim6: Level 3 multihoming shim protocol for IPv6. IETF network working group," 2009.

[4] E. L. Følstad and B. E. Helvik, "Reliability modelling of access point selection and handovers in heterogeneous wireless environment," in Design of Reliable Communication Networks (DRCN), 2013 9th International Workshop on the, 4–7 March 2013.

[5] ——, "Optimizing service continuity in a multi operator multi technology wireless environment," in Design of Reliable Communication Networks (DRCN), 2013 9th International Workshop on the, 4–7 March 2013.

[6] E. W. Dijkstra, "A note on two problems in connexion with graphs," Numerische Mathematik, vol. 1, pp. 269–271, 1959, 10.1007/BF01386390.

[7] R. Bhandari, Survivable networks: algorithms for diverse routing. Kluwer Academic Pub, 1999.

[8] J. Holland, "Adaptation in natural and artificial systems. 1975," Ann Arbor, MI: University of Michigan Press and, 1992.

[9] C. R. Reeves, "Genetic algorithms," in Handbook of Metaheuristics, ser. International Series in Operations Research & Management Science, M. Gendreau, J.-Y. Potvin, and F. S. Hillier, Eds. Springer US, 2010, vol. 146, pp. 109–139.

[10] B. Dengiz, F. Altiparmak, and A. Smith, "Efficient optimization of all-terminal reliable networks, using an evolutionary approach," Reliability, IEEE Transactions on, vol. 46, no. 1, pp. 18 –26, mar 1997.

[11] C. W. Ahn and R. Ramakrishna, "A genetic algorithm for shortest path routing problem and the sizing of populations," Evolutionary Computation, IEEE Transactions on, vol. 6, no. 6, pp. 566 – 579, dec 2002.

[12] J. J. Grefenstette, "Optimization of control parameters for genetic algorithms," Systems, Man and Cybernetics, IEEE Transactions on, vol. 16, no. 1, pp. 122 –128, jan. 1986.

[13] B. Dengiz, F. Altiparmak, and A. Smith, "Local search genetic algorithm for optimal design of reliable networks," Evolutionary Computation, IEEE Transactions on, vol. 1, no. 3, pp. 179 –188, sep 1997.

[14] I. Wolfram Research. (2011) Documentation centre.