

iPatch: A Many-Objective Type-2 Fuzzy Logic System for Field Workforce Optimisation

Andrew Starkey, Hani Hagrass, *Fellow, IEEE*, Sid Shakya, Gilbert Owusu

Abstract— Employing effective optimisation strategies in organisations with large workforces can have a clear impact on costs, revenues, and customer satisfaction. This is particularly true for organisations that employ large field workforces, such as utility companies. Ensuring each member of the workforce is fully utilised is a challenging problem as there are many factors that can impact the organisation's overall performance. We have developed a system that optimises to make sure we have the right engineers, in the right place, at the right time, with the right skills. This system is currently deployed to help solve real-world optimisation problems, which means there are many objectives to consider when optimising, and there is much uncertainty in the environment. The latest version of the system uses a multi-objective genetic algorithm as its core optimisation logic, with modifications such as Fuzzy Dominance Rules (FDRs), to help overcome the issues associated with many-objective optimisation. The system also utilises genetically optimised type-2 fuzzy logic systems to better handle the uncertainty in the data and modelling. This paper shows the genetically optimised type-2 fuzzy logic systems producing better results than the crisp value implementations in our application. We also show that we can help address the weaknesses in the standard NSGA-II dominance calculations by using FDRs. The impact of this work can be measured in a number of ways; productivity benefit of £1million a year, the reduction of over 2,500 metric tonnes of CO₂ and a possible prevention of over 100 serious injuries and fatalities on the UK's roads.

Keywords— Type-2 fuzzy logic, many-objective, multi-objective, genetic algorithms, workforce optimisation, fuzzy dominance

I. INTRODUCTION

LARGE organisations can suffer many inefficiencies in their daily operation if not appropriately addressed. This is especially true of organisations with large mobile workforces. Failure to tackle these inefficiencies can cause negative side effects, such as higher travel costs, lower capacity to service new customers, slower response times and higher environmental impacts. The more complex the organisation, the more difficult it is to identify and resolve these issues on a regular basis [1][2].

Andrew Starkey is with the Computational Intelligence Centre, School of Computer Science and Electronic Engineering, University of Essex, Wivenhoe Park, Colchester, CO43SQ, UK. (e-mail: astark@essex.ac.uk)

Hani Hagrass is with the Computational Intelligence Centre, School of Computer Science and Electronic Engineering, University of Essex, Wivenhoe Park, Colchester, CO43SQ, UK. (e-mail: hani@essex.ac.uk)

Sid Shakya is with EBTIC, Khalifa University, Abu Dhabi, UAE (e-mail: sid.shakya@kustar.ac.ae)

Gilbert Owusu is with the Business & Operational Transformation Practice, British Telecom, Adastral Park, Ipswich, UK (e-mail: gilbert.owusu@bt.com)

An example of a large organisation with a large mobile workforce is a utility company. These organisations deal with the complex infrastructure that has built up over decades, meaning there is usually a large degree of maintenance and upgrades that need to be carried out by a human workforce. The workforce has to travel to each part of the infrastructure network to carry out their tasks. The effective allocation of these tasks to each engineer requires certain objectives to be considered; to minimise travel time and to maximise completed tasks. Effective allocation of tasks to a mobile workforce can be thought of like the travelling salesman problem [3]. However, unlike the travelling salesman problem (TSP), there isn't just one travelling entity, there can be thousands. As a result, any small inefficiencies have a noticeable impact on the utilisation of the workforce, fuel costs and emissions.

A traditional way of managing this problem is to divide the geography into regions and sub-regions and install a hierarchical management structure of regional managers, sub-regional managers, team leaders, etc. The managers will be responsible for all lower levels of their allocated geography. The task of managing the infrastructure, task allocation and engineers is a very complex process. There are many more aspects to consider. For example, engineers can only be allocated to tasks they are qualified to complete. More importantly, the geographical areas themselves have to be intelligently designed, so that they are balanced for the operation of the field teams. They cannot be separated by large geographical obstacles, such as rivers, and should contain a good balance of tasks and engineers.

This produces one of the primary optimisation problems; the design of these geographical Working Areas (WAs). These working areas form the boundaries in which the engineering teams should work and the geographic area for which the managers are responsible. Each of the WAs is made up of a collection of Service Delivery Points (SDPs) which connect to local properties (household and commercial). An SDP contains tasks to be completed by the engineers and will include different types, and amounts of work, depending on the SDPs location.

With geographies covering hundreds of thousands of square kilometres, and tens of thousands of engineers needed to service these areas, the problem space is vast. We have implemented meta-heuristic optimisation algorithms to tackle the utility company's organisational inefficiencies.

One of the primary objectives of the optimisation process is to increase the amount of work completed. This leads to tasks being completed sooner, increasing customer satisfaction and increasing the capacity of the workforce.

Secondary benefits include lower fuel costs and lower CO₂ emissions due to minimised travel. Lastly, in times of unexpected demand (such as tasks due to extreme weather conditions), the organisation is more robust, reducing the stress put onto the workforce when repairing lost utilities is seen as urgent.

As well as using meta-heuristics to search for optimal solutions, we are using Fuzzy Logic Systems (FLSs) within the optimisation process to make better decisions in the WA designs and evaluations. The use of fuzzy systems in real-world problems have been shown to improve the outcomes in many real-world applications, such as in breast cancer patient classification [4]. Also, FLSs are particularly useful for handling the uncertainties in estimated distances, travel times and estimated task length [5][6]. We are using fuzzy logic to tackle the many-objective problem by improving the meta-heuristics ability to identify better solutions.

This paper presents the work completed on iPatch, BT’s many objective type-2 fuzzy logic system for field workforce optimisation. Over the first two years of deployment, this application has *increased productivity by 0.5% across the mobile workforce and reduced fuel consumption by 2.9%. This has led to a productivity benefit of £1million a year and a saving of over £200K a year in fuel costs.*

Secondary benefits include *an estimated reduction of 2,500 metric Tonnes of CO₂ and reduced serious traffic casualties or fatalities by over 100.*

The rest of the paper is organised as follows: Section II presents an overview of field workforce optimisation. In section III, we provide a brief overview of genetic algorithms for multi-objective problems and an overview of type-2 fuzzy systems.

Section IV presents a description of how the system functions. Section V describes how we genetically optimise our fuzzy systems. Section VI reports on the experiments and the results obtained from them. Section VII describes our results for the FDRs. Section VIII talks about the real-world impact that iPatch has had. Finally, Section IX presents the conclusions and potential future work.

II. OVERVIEW OF FIELD WORKFORCE OPTIMISATION PROBLEMS

Any organisation can suffer inefficiencies if effective planning and organisation strategies are not in place. The larger the organisation is, and the longer the organisation operates without optimising their business, the faster these inefficiencies will increase and impact the business.

The subject of Workforce Management (WFM) is broad [7], and optimisation strategies will differ between workforce types. A field workforce is one that is mobile and will travel to many locations throughout the day. Typical examples of field workforces include travelling salespeople, utility companies and delivery companies. The travelling salesman problem (TSP) directly relates to field workforce optimisation, where one of the primary objectives is to minimise the amount of travelling any member of the workforce has to do during a day.

A. Overview of Work Area Optimisation

One way to minimise travel and increase utilisation of a field workforce is to split up the geography into different territories for teams to manage. This geographical division can form a management hierarchy by grouping the territories together. These groupings can then have their own TSP modelled and optimised. This reduces the size and complexity of the problem.

For our organisation’s hierarchy, the lowest level is known as a Working Area (WA) The WAs are made up of clusters of Service Delivery Points (SDPs). Each SDP services local properties and generates demand for the mobile engineering workforce.

Fig. 1a shows how part of the UK might be divided into regions and Fig 1b shows a sub-region, which is divided into five WAs. SDPs are shown as dots within each of the WAs. Clustering the SDPs together into the best configurations is the central part of geography optimisation for our system.

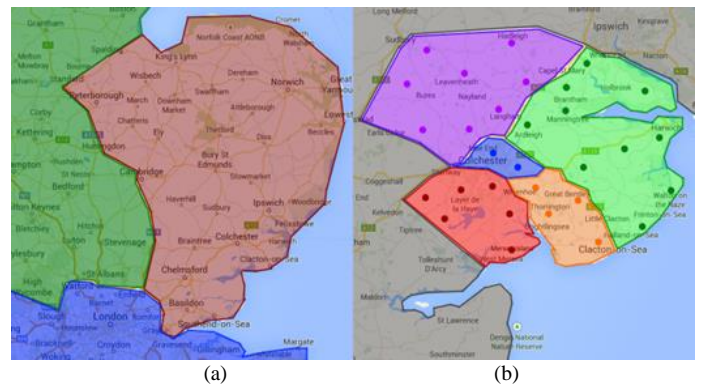


Fig. 1. a) Regional Areas. b) WAs within a Sub-Region.

B. Overview Of Resource Optimisation

Optimising the geography is only one mechanism we can employ to improve the utilisation of a workforce. Another way is to optimise the available resources skills and to optimise the teams the resources are assigned to, without changing the geographical structures of the working areas. We call it resource optimisation.

There is a meta-heuristic for selecting the best combination of resources to train (or ‘upskill’) and for the optimal team allocation for all engineers.

The first of these optimisations, the upskilling, decides which engineers to choose to go on training courses, given any constraint. So if a regional manager only has the budget to send ten engineers on a training course in a given month in their area, then out of the few hundred eligible engineers, which would return the most benefit and not violate any business constraints?

The second optimisation available is to choose n number of engineers to move to another team such that the skill is balanced for both teams. This is useful for engineers on the borders of working areas.

This paper will not go further into this part of the system, but more information can be found in our previous publication on this topic [8].

C. Objectives and Constraints

For our particular workforce optimisation strategy, we have five potential objectives. If all of five objectives are used it qualifies our optimisation problem as many-objective. A many objective problem is one with four or more objectives [9],[10].The objectives of our optimisation are as follows:

Maximise Coverage: This is the basic measure of work that is expected to be completed by the engineers. This is measured as a percentage of total completed work. Equation (1) represents the sum of all engineers expected completed work. Over the region's total work (R_{TW}). The region contains all the WAs being optimised. C_i is an individual's completed work, measured in hours, n is the number of engineers. R_{TW} is measured in hours.

$$C = \left(\frac{1}{R_{TW}} \right) \sum_{i=1}^n C_i \quad (1)$$

Minimise Travel: Minimising the amount an engineer travels increases the available productive time for each engineer. Reducing travel also reduces costs. Minimising travel conflicts with maximising coverage as an engineer will usually be required to travel to each task. As coverage increases, travel also increases. In Equation (2) this is represented as the sum of all engineers' travel distance (E_{TD}) divided by the total number of engineers (E) representing travel as an average for the workforce. Travel is measured in kilometres.

$$\sum E_{TD} / \sum E \quad (2)$$

Maximise Utilisation: Unutilised time is when the engineer is idle or travelling, and hence we want to maximise the utilisation of the workforce. Equation (3) shows the sum of engineer completed work (E_{CW}) divided by the engineer's available time (E_{AT}), this is then divided by the total number of engineers (E).

$$\left(\sum E_{CW} / E_{AT} \right) / \sum E \quad (3)$$

Minimise Area Imbalance: WAs should have an even distribution of demand. This will lead to smaller WAs for urban areas, and larger WAs for rural areas. Area balancing is the difference between the largest (WA_L) and smallest (WA_S) regarding hours of work, shown in Equation (4).

$$WA_L - WA_S \quad (4)$$

Minimise Team Imbalance: WAs should have evenly balanced teams. Team Balance is the difference between the largest (T_L) and smallest (T_S) teams, shown in Equation (5) and measured in the number of people.

$$T_L - T_S \quad (5)$$

III. A BRIEF OVERVIEW OF GENETIC ALGORITHMS AND TYPE-2 FUZZY LOGIC SYSTEMS

A. Multi-Objective Genetic Algorithms

Genetic Algorithms (GAs) are a widely used meta-heuristic for real-world optimisation problems [11], [12], [13]. It is population-based, where each new generation of the population aims to contain stronger solutions to a given problem than the previous generation. A more in-depth description on GAs is given in Section II of the Supplementary Material.

Traditionally fitness functions are created to evaluate how good a solution is compared to others in the same population set. If there are multiple objectives, the objective values will form parts of this function depending on the type of objective (i.e. minimisation or maximisation).

To highlight why this is not always a suitable way of evaluating a multi-objective solution consider the following example. A problem has three objectives; A, B and C the current solution to this problem has the values for these objectives as 5, 5 and 6 respectively. Objectives A and B are maximisation objectives, and C is a minimisation objective. Finally, we are told B and C are conflicting, and an increase in B increases C with an exponential relationship. The resultant fitness function is shown in Equation (6) which also shows the example fitness value of 4.17 for the original solution.

$$Fitness = \frac{A \times B}{C} = 4.17 \quad (6)$$

Table 1 shows possible solutions that could be generated by the GA. Every solution presented in Table 1 is deemed superior to the original solution, based on the fitness value. No solution optimises in all three objectives when compared to the original solutions objective values.

TABLE I. EXAMPLE SOLUTIONS FOR A MULTI-OBJECTIVE PROBLEM USING A FITNESS FUNCTION

Possible Solutions	Objective A (Maximise)	Objective B (Maximise)	Objective C (Minimise)	Fitness Score
Original	5.00	5.00	6.00	4.17
1	6.00	6.00	6.60	5.45
2	6.50	7.00	7.70	5.91
3	4.95	2.90	3.19	4.50
4	4.70	2.70	2.97	4.27

The example shows solutions can be generated by the GA and fail in one of the desired objectives, yet be considered a strong solution. This affects the output of the algorithm and also the parent selection stage.

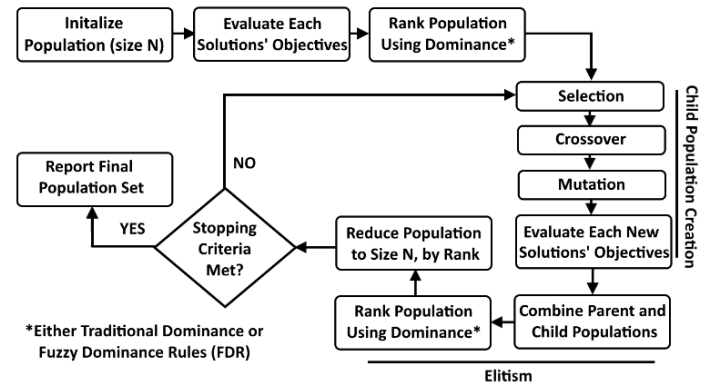


Fig 2. The flow of a MOGA.

Multi-Objective Genetic Algorithms (MOGAs) share much of the same process as single objective GAs. However, the comparisons between solutions are different. The MOGA we are specifically referring to is NSGA-II [14]. NSGA-II will use dominance rules to evaluate and compare solutions, instead of a fitness function. Fig 2. Shows the typical process of a MOGA. This process starts by generating a random set of n solutions to

create the initial population. Each solution is evaluated and then ranked using dominance. The method of dominance is outlined below.

Selection, Crossover and Mutation are processed to generate new solutions for the child population set. The children are added to the original population, and then dominance is used again to rank the population. The population will then be reduced from size $2n$ to size n by removing the n weakest solutions.

The process will return to the Selection, Crossover and Mutation stages if the stopping criteria are not met, else the dominant solutions will be reported to the user.

It should be noted that where dominance is used Fuzzy Dominance Rules can also be used (as marked by the * in Fig. 3). This is to be discussed in Section VII.

Dominance has two conditions associated with it when assessing if one solution is superior to another. The following description of these conditions is to determine if the first solution (A) dominates the second (B):

1. Solution A has no objective value that is worse than the respective objective value in B.

2. Solution A has at least one objective value that is better than the respective objective value in B.

Solution A will be considered the superior solution if both conditions are met, thus solution A would ‘dominate’ solution B. The domination count for each solution is calculated by checking these dominance conditions for each solution against every other solution in the population set. By using domination count, a sorting algorithm can then arrange the solutions into sets, in order from the best (with a count of 0) to worst (with the highest count attained by the domination counting algorithm). The best grouping (where the domination count is 0) is known as the Pareto front, or Pareto set [15]. The next front is made up of solutions with the next best domination count (which may not necessarily be 1). This process continues until all solutions are grouped into their respective fronts.

B. Hypervolume Indicator

Once a MOGA produces a Pareto front we can measure the hypervolume of the shape [16], where the shape is produced by the Pareto solutions and reference points. The reference points represent the maximum value that a minimisation objective could be (and vice versa). This is illustrated in Fig. 3.

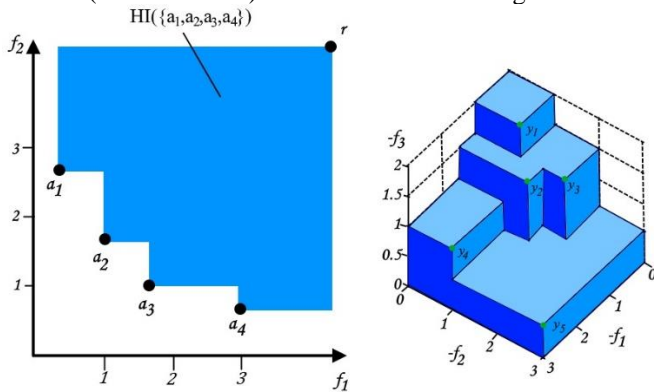


Fig 3. Hypervolume Indicator in two dimensions for a set $A = \{a_1, \dots, a_4\} \subset \mathbb{R}^2$ (left) and in three dimensions for a set $Y = \{y_1, \dots, y_5\} \subset \mathbb{R}^3$ (right).[17]

We use the hypervolume of the Pareto fronts generated by our system to compare the different methods we are testing. As we have some maximisation objectives, we will convert the maximisation objective values to negative (i.e. 10 will become -10) and set the reference point of these objectives to 0. As we know in this case, 0 for maximisation objective is the worst possible outcome. In our application, there are up to five dimensions.

C. Type-2 Fuzzy Logic Systems

Fuzzy Logic Systems (FLSs) have been credited with handling uncertainty and imprecision. However, the vast majority of the FLSs were based on type-1 fuzzy logic systems which cannot fully handle or accommodate for the uncertainties associated with changing and dynamic environments, such as those found in commercial applications, like the one discussed in this paper. Type-1 fuzzy sets handle the uncertainties associated with the FLS inputs and outputs by using precise and crisp membership functions [18]. Once the type-1 membership functions have been chosen, all the uncertainty disappears, because type-1 membership functions are totally precise [18], [19].

Type-2 fuzzy sets have the potential to handle the high levels of uncertainty associated with real-world environments (for example, travel times) and give a good performance as a result.

A fuzzy membership function characterises a type-2 fuzzy set, i.e. the membership value (or membership grade) for each element of this set is a fuzzy set in $[0,1]$, unlike a type-1 fuzzy set where a membership grade is a crisp number in $[0,1]$ [18]. The membership functions of type-2 fuzzy sets are three-dimensional and include a Footprint of Uncertainty (FOU), it is the third-dimension of type-2 fuzzy sets and the footprint of uncertainty that provide additional degrees of freedom that make it possible to directly model and handle uncertainties [18], [19], [20], [21].

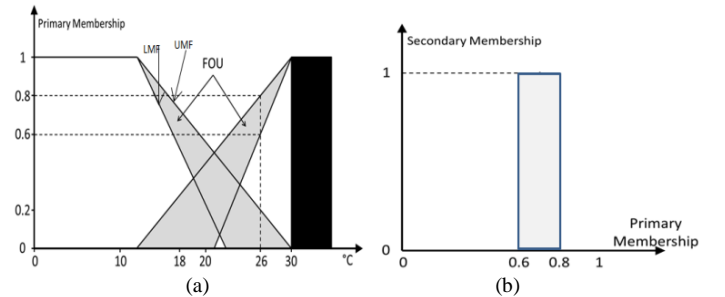


Fig 4. a) An interval type-2 fuzzy set - primary membership function. b) An interval type-2 fuzzy set secondary MF at a specific point x' .

As shown in Fig 4a, the Interval Type-2 (IT2) fuzzy set \tilde{A} can be represented in terms of the Upper Membership Function (UMF) (denoted by $\bar{u}_{\tilde{A}}(x), \forall x \in X$) and the Lower Membership Function (LMF) (denoted by $\underline{u}_{\tilde{A}}(x), \forall x \in X$) as follows:

$$\tilde{A} = \int_{x \in X} \left[\int_{u \in [\underline{u}_{\tilde{A}}(x), \bar{u}_{\tilde{A}}(x)]} 1/u \right] / x \quad (7)$$

The UMF and LMF are bounds for the $FOU(\tilde{A})$ of an IT2 fuzzy set \tilde{A} . As shown in Fig 4b, in an IT2 fuzzy set the secondary membership function is equal to 1 for all the points in the primary membership for $\forall x \in X$.

In type-2 FLSs, the crisp inputs are fuzzified to input type-2 fuzzy sets which are fed to the inference engine which maps the input type-2 Fuzzy sets to output type-2 fuzzy sets using the rule base. The type-reducer processes the output set in the type reduction section which generates a type-1 output set. In this paper, we use the Centre of Sets type-reduction, shown in Equation (8), as it has a reasonable computational complexity that lies between the computationally expensive centroid type-reduction and the simple height and modified height type-reductions which have problems when only one rule fires [18].

$$Y_{\text{cos}}(x) = [y_l, y_r] = \int_{y^1 \in [y_l, y_r]} \dots \int_{y^m \in [y_l^m, y_r^m]} \int_{f^1 \in [f^1, f^1]} \dots \int_{f^M \in [f^M, f^M]} 1 / \frac{\sum_{i=1}^M f^i y^i}{\sum_{i=1}^M f^i} \quad (8)$$

IV. THE MANY-OBJECTIVE GENETIC TYPE-2 FUZZY LOGIC SYSTEM FOR WORKFORCE OPTIMISATION

The system that has been built to optimise Working Areas (WAs) is illustrated in Fig. 5. The system contains many parameters; including the GA's parameters, the optimisation goals and configurations for the integrated fuzzy systems. These parameters are currently entered by the user depending on their current requirements. The confirmation step allows a final check before time and resource are allocated to the task.

The first setting relates to the fuzzy systems and whether the fuzzy systems to be used in the optimisation should be optimised for the current geography the system is being applied to.

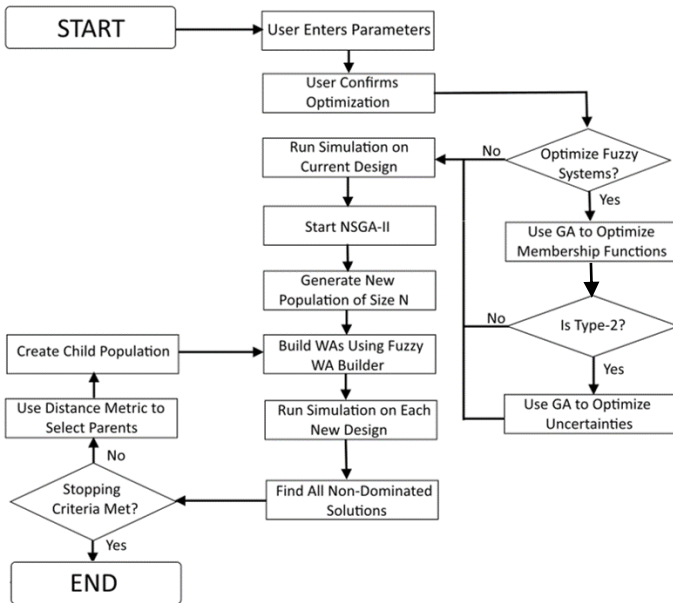


Fig. 5. The Genetic Type-2 Fuzzy Logic Based System for Mobile Field Engineer WA Optimisation.

If the fuzzy systems are to be optimised, a GA will run to tune the membership functions in the fuzzy sets to be used. If the user has selected that they wish for Type-2 fuzzy systems to be used,

another GA will optimise the footprints of uncertainty associated with the newly optimised membership functions. A brief overview of how the GA optimises the fuzzy systems can be found in Section V.

We have the option of using type-1 and type-2 fuzzy logic in our system so that we can run both variations and compare the outputs. This forms part of our experiments as we examine which is best.

Once the pre-optimisation stage has been completed the system will run the simulation on the current WA designs. The simulation estimates the productivity and cost of a typical day, based on the geographical design and available engineers.

The simulation gives us the values for the coverage, travel and utilisation objectives. Then the balance of the WAs and teams is calculated by finding the largest and smallest WAs and teams within the design. These values for the original design are crucial to the calculation of the distance described in Section IV C.

The NSGA-II algorithm will start once the original designs objective values have been calculated. NSGA-II will randomly generate the first population set. From the first population set, the WA builder will generate new WAs from the chromosome of each solution. Each gene in the chromosome holds a reference to an SDP. These act as central nodes to cluster the remaining SDPs to, thus constructing new WAs. More information on the WA builder and its fuzzy system can be found in Section IV B. The number of genes in a solution's chromosome is equal to the number of WAs in a design.

Every new design created by NSGA-II will have the simulation performed on it, and both of the balancing objective values will be calculated to get the five objective values for each solution. These objective values will then allow NSGA-II to perform its dominance calculation to generate the Pareto front.

If the stopping criteria are not met, then NSGA-II will continue to the evolution stage (Parent Selection, Crossover and Mutation). We have increased NSGA-II's ability to identify strong solutions by implementing a distance metric, see Section IV C for further details.

With many-objective problems, the Pareto front may become overpopulated, and this can be overwhelming for the user. The distance metric is also used to choose the best solution from the Pareto front. From the user's point of view, the system still suggests a single solution, but also has the ability to cycle through additional solutions that may better suit their needs.

A. Fuzzy Simulation

The simulation used in the system uses data that contains uncertainty. Using fuzzy systems to handle this uncertainty can improve the realism of the simulation and hence the results transfer into the real-world to a higher degree. [22] demonstrates a fuzzy simulation for the construction industry. Uncertainty in our data includes the length of tasks, travel distances and times, the quantity of tasks available and availability of engineers.

As an example, when deciding to pick up a task in the simulation, an engineer might decide to compare the distance to two different tasks. If one task is 25 minutes away and the other is 27 minutes away, the closest shouldn't necessarily win based on these crisp values, particularly because these values do not reflect ever-changing traffic conditions.

TABLE II. TASK ALLOCATION RULE BASE

Distance to task	Tasks at SDP	Chance of Choosing SDP
Low	Low	Average
Low	High	High
High	Average	Low

The design of the fuzzy system for the simulation also looked at the amount of work in the SDP, as this varies from day to day. Some example rules of how these two fuzzified simulation components affect the decision making of the simulated engineers are given in Table II. The ‘Tasks at SDP’ are tasks only applicable to the engineer making the decision. This decision making fuzzy system used the Centre of Sets type-reduction, shown in Equation (8) for the reasons outlined. A more in-depth description of this system is given in Section V of the Supplementary Material.

B. Fuzzy WA Builder

Each geographical solution is created by a neighbourhood based clustering algorithm. Each gene in a solution represents an SDP to act as a centre point to each cluster. We illustrate the clustering process in Fig 6. Fig 6a shows three SDPs selected as the centre points. Fig 6b shows the immediate neighbours being added; figure 6c shows the next few layers of SDPs being added. Lastly, Fig 6d shows the final design created by the three SDPs, selected by the GA. If an SDP neighbours more than one cluster we use a more intelligent decision-making process to decide which area it should be added to.

We assess how far away by travel time the SDP is from the centre point, how much work is in the SDP to be added and how much work is already in the current generated WA. For example, if one WA is already heavy with work and the SDP to be added has many tasks, it may not be a good idea to add that SDP. The fuzzification comes in when looking at the neighbouring SDPs distance away from the current WA, the size of that SDP in the amount of work and the size of the current WA the SDP may be added to.

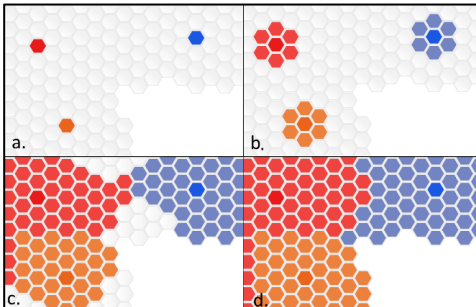


Fig. 6. An example of the SDPs being clustered by their neighbours.

Table III gives some example rules for this fuzzy system. As with the previous system a human expert designed the rules and fuzzy sets.

TABLE III. WA BUILDER RULE BASE

WA Size	Distance To	SDP Size	Consequence
Average	Small	Small	Add
Average	Large	Small	Add
Average	Average	Large	Don't Add
Large	Small	Average	Don't Add

This decision making fuzzy system used the Centre of Sets type-reduction, shown in Equation (8) for the reasons outlined.

A more in-depth description of this system is given in Section IV of the Supplementary Material, or our previous work [23].

C. The Implemented Distance Metric

To help identify strong solutions in the population, we have implemented a distance metric that increases when maximisation objectives are increased and increases when minimisation objectives are decreased, when compared to the original WA design. The distance metric for our objectives is shown by Equation (7). The greater the distance from the original solution design the better the new solution is.

$$DIST = \left(\frac{C_S - C_O}{C_O} - \frac{T_S - T_O}{T_O} + \frac{U_S - U_O}{U_O} - \frac{AB_S - AB_O}{AB_O} - \frac{TB_S - TB_O}{TB_O} \right) \quad (7)$$

$$DIST = \frac{New - Original}{Original} \quad (8)$$

Equation (7) shows the coverage for the new solution (C_S) and the original solution (C_O). The travel value given by the new solution (T_S) and the travel value given by the original solution (T_O). The utilisation given by the new solution (U_S) and the utilisation given by the original solution (U_O). The area balance given by the new solution (AB_S) and the area balance given by the original solution (AB_O). Finally, the team balance given by the new solution (TB_S) and the team balance given by the original solution (TB_O).

Each objective in equation (7) calculates a local objective distance using Equation (8). The original value subtracts the new value for that objective, divided by the original.

The distance metric used by the application will depend on the objectives being optimised. If only three objectives are being optimised, then only those three objectives will contribute to the distance metric. In our system, we combine crowding distance with the distance metric to choose solutions as the tie-breaker, where the distance metric has a dominant weighting.

D. Dominance in Many-Objective Problems

Many objective problems are described as those with four or more objectives [9] [10]. The more objectives there are, the more likely that the mentioned dominance rules in section III-A will not be sufficient to distinguish between good solutions. Thus, the Pareto front will become saturated with solutions (sometimes containing all solutions) making it very difficult to choose parents in the selection stage of the GA.

The problem stems from the first rule; that no objective can be worse. Consider the results in Table IV, where each of the five objectives are minimisation objectives.

TABLE IV. DOMINANCE IN MANY-OBJECTIVE PROBLEMS: EXAMPLE I

Solution No	Objective 1 (min)	Objective 2 (min)	Objective 3 (min)	Objective 4 (min)	Objective 5 (min)	Solutions Dominated
1	3	6	8	4	7	0
2	2	5	5	4	8	0
3	2	6	1	5	1	0
4	1	1	1	5	2	0
5	8	1	1	1	1	0

In Table IV, solution 4 does a very good job of minimising all objective, except objective 4. This objective has been sacrificed for all others. An expected outcome with conflicting objectives.

The same could be said of solution 5. These are clearly good solutions, however because of the rule stating no objective can be worse, these solutions fail to dominate the clearly weaker ones. Selection pressure does not consider strong solutions because of this; it has to rely on weaker or secondary selection pressures such as crowding distance. The problem is exaggerated in Table V.

Table V shows another situation where we have five solutions that do not dominate each other. However, to any human solutions, 4 and 5 are clearly better. Solutions 1, 2 and 3 have failed in the majority of the objectives, but under dominance, they are good candidates for selection as parents.

To address this problem, we will use our proposed Fuzzy Dominance Rules [24]. Briefly, this is the introduction of a fuzzy system in place of the standard dominance rule check. Each objective value is fuzzified and then compared. The membership functions for this FLS are proportional to the values being compared. For example, 10% grace value on objective 4 when comparing solutions 2 and 4 from Table V would mean solution 4 could have a value of 4.4 and the condition of ‘no objective worse’ would still be met.

TABLE V. DOMINANCE IN MANY-OBJECTIVE PROBLEMS: EXAMPLE II

No.	Objective 1 (min)	Objective 2 (min)	Objective 3 (min)	Objective 4 (min)	Objective 5 (min)	Solutions Dominated
1	3.0	100.0	800.0	4.0	70.0	0
2	2.0	100.0	50.0	4.0	80.0	0
3	2.0	410.0	1.0	50.0	1.0	0
4	1.0	1.0	1.0	4.1	1.1	0
5	3.1	1.0	1.0	1.0	1.0	0

The Fuzzy Dominance Rules for many-objective dominance comparisons operates like a traditional type-1 fuzzy system. The crisp output value of this FDR system will decide if one solution dominates another by comparing the output values. This replaces the dominance comparison used in NSGA-II.

The input sets to the system are defined by the objectives being compared. Fig. 7 illustrates how the input sets are generated. A represents the objective value, and T represents the tolerance.

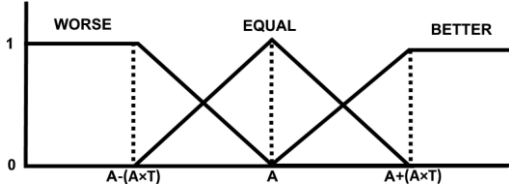


Fig. 7. Input Fuzzy Sets generated from objective value A, with a tolerance level of T

V. THE GENETICALLY OPTIMISED FUZZY SYSTEMS

Membership functions in fuzzy sets that are generated by a human expert are subject to noise and uncertainty. Using an optimisation algorithm to tune these membership functions can improve the fuzzy system’s performance [25]. When a type-2 fuzzy system is used, there is also the opportunity to optimise the footprints of uncertainty (FOU) associated with these membership functions.

The fuzzy system described in Section IV-A is comprised of two inputs and one output. The Distance input is represented by three fuzzy sets while the Task input is represented by five fuzzy sets and the fuzzy system output is represented by five fuzzy sets. The system described in IV-B has three inputs and one output.

The three inputs are represented by three fuzzy sets and the output is presented by two fuzzy sets. The number of fuzzy sets for each input/output was decided by an industry expert to allow them to understand how the system was built. We always started with three fuzzy sets for all input sets, then this number was adjusted to improve the performance of the system yet retain interpretability. These systems are further described in [23].

We employed Genetic Algorithms (GAs) using real value representation to optimize the parameters of the fuzzy sets Membership Functions (MFs). The genes of each GA chromosome will represent the points of each MF along the x-axis.

Fig. 8 shows an example of a chromosome for the parameters of two type-1 fuzzy sets MFs. Each MF is made up of four points along the x-axis. The number of genes is $4n$ where n is the number of the fuzzy sets. Fig 9 is an example chromosome for the type-2 FOUs, which will be used if a type-2 system is to be optimised. The number of genes here is $2 * \frac{4n}{2}$ as only half of the points in the MF defined by Fig. 8, are Membership Function Base Points (MFBBPs).

However, as each base point requires two uncertainty values, for the left and right membership functions, then the number of genes to calculate the FOU values will be $4n$.

Each gene in Fig. 9 represents the uncertainty percentage associated with the MFBBP values of the type-1 membership functions. The left and right uncertainty values in the membership functions for each base point, are calculated using pairs of genes. For example, 20 is the first MFBBP from Fig. 8 is denoted by *. The Left Membership Function Uncertainty value (LMFU) and the Right Membership Function Uncertainty value (RMFU) associated with this first base point are also denoted by * in Fig. 9. The calculation for the LMF base point is given by Equation (9), and RMF base point is given by Equation (10).

Equations (11) and (12) illustrate how the values are calculated using the values from Fig. 8 and Fig.9. Fig. 10 show the resulting type-2 fuzzy set generated by these chromosomes.



Fig. 8. Real-value chromosome for the parameters of two type-1 fuzzy sets membership functions



Fig. 9. Real-value chromosome for percentage uncertainty associated with the type-2 fuzzy sets

$$LMF \text{ Base Point} = MFBBP - (MFBBP \times LMFU) \quad (9)$$

$$RMF \text{ Base Point} = MFBBP + (MFBBP \times RMFU) \quad (10)$$

$$LMF \text{ Base Point} = 20 - (20 \times 0.1) = 18.0 \quad (11)$$

$$RMF \text{ Base Point} = 20 + (20 \times 0.15) = 23 \quad (12)$$

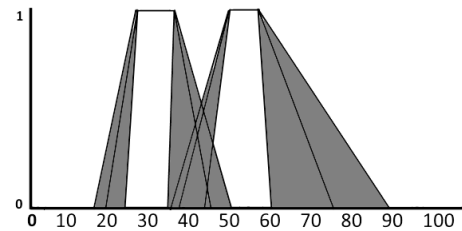


Fig. 10. Resulting Type-2 Membership Functions from Example

VI. EXPERIMENTS AND RESULTS

We have a number of experiments that aim to show improvements to our modified NSGA-II algorithm. The first experiment utilises the type-1 fuzzy versions of the fuzzy simulator and fuzzy WA builder as outlined in section IV. The second experiment replaces the fuzzy systems with type-2 FLSs. The type-2 fuzzy system was described in Section III C. The third experiments utilises the type-2 FLSs but has a short optimisation with a GA before the WA optimisation takes place, as described in section V. Finally we will expand our discussion in Section VII as the remaining experiments detail our solution to the many-objective optimisation issues brought about by the standard NSGA-II dominance rules as detailed in section III A, III B and IV D.

TABLE VI. ORIGINAL RESULTS

Coverage (%)	Travel (km)	Utilisation (%)	Area Balance (Hours)	Team Balance (people)
76.12	26.50	68.15	354.65	71

The experiments start by choosing a single geographical area to optimise. We evaluate the current design in this area to get our benchmark (or ‘Original’) objective values for this area. These values can be found in Table VI.

As our first aim is to show that the introduction of FLSs improves our system, but we will only choose three objectives to include. Else, the problems associated with many-objective optimisation may impact the results and give a false impression of the fuzzy systems ability to improve our results. For the first four sets of results only Coverage, Travel and Area Balancing will be used as our objectives.

Each experiment will run the optimisation 30 times, giving 30 unique seed values each time. Each experiment will use the same 30 unique seed values to reduce the elements of randomness further. Each run will give a Pareto front where we will use the discussed hypervolume metric, from Section III B, to evaluate the Pareto fronts. The reference points for the three objectives will be 0, 100 and 1000 for Coverage, Travel and Area Balance. We will refer to our modified NSGA-II as ‘the MOGA’. All the hypervolumes from the experiments are shown in the Hypervolume Summary Table, Table VII. Table VII shows the hypervolume set for the MOGA as **M**, the introduction of type-1 fuzzy systems gives us the hypervolume set noted by **T1**. The

upgrade to type-2 systems gives us the hypervolume set noted by **T2**. Finally, the hypervolume set given by the MOGA with genetically optimised type-2 fuzzy logic systems is denoted by **OT2**. For each set, the results are shown on two rows, runs 1-15 are on the first row and runs 16-30 are on the second. We can plot a Pareto front result from each of the hypervolume sets for a visual comparison. Figs 11-13 show different perspectives of the same four Pareto fronts, in a three-dimensional environment.

These Pareto fronts were taken from each method’s final result from the same seed. These graphs clearly show the conflicting relationship between coverage and travel. They also highlight a positive correlation showing more balanced WA designs lead to higher levels of task coverage.

If we look at the average (Avg.) of the 30 runs in Table VII for each hypervolume set, we can see that best average hypervolume was achieved by OT2, followed by T1 & T2, followed, finally, by M. This is a similar pattern seen in our earlier work [26] where the distance metric was used in place of the hypervolume. For each seed value, the winning result has been bolded. OT2 wins (or draws) in half of all cases and wins almost twice as often as any other method. OT2 also finds the best result overall, 0.75. Though OT2 does not win with all seeds, we can be confident that the OT2 method will produce stronger results more frequently.

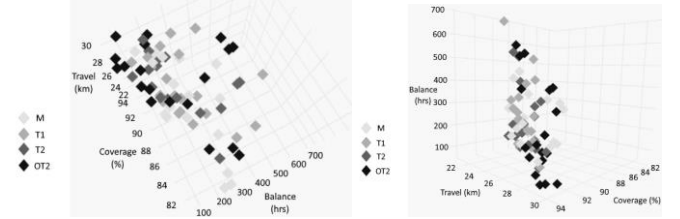


Fig. 11. 3D plot of Pareto fronts (1) Fig. 12. 3D plot of Pareto fronts (2)

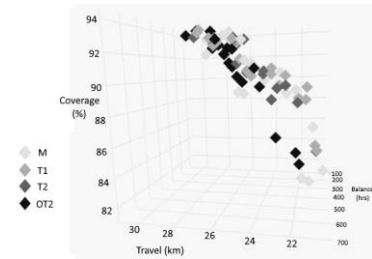


Fig. 13. 3D plot of Pareto fronts (3)

TABLE VII. HYPERVOLUME SUMMARY TABLE

																Avg.
M	0.63	0.73	0.69	0.62	0.65	0.65	0.68	0.71	0.68	0.63	0.68	0.67	0.69	0.67	0.74	0.67
	0.70	0.71	0.64	0.63	0.64	0.67	0.65	0.64	0.70	0.71	0.69	0.62	0.70	0.71	0.72	
T1	0.72	0.67	0.67	0.70	0.72	0.59	0.67	0.67	0.69	0.66	0.69	0.70	0.67	0.69	0.68	0.68
	0.66	0.69	0.67	0.74	0.66	0.65	0.67	0.69	0.71	0.63	0.67	0.68	0.65	0.66	0.69	
T2	0.62	0.66	0.70	0.71	0.71	0.71	0.69	0.70	0.71	0.71	0.64	0.66	0.66	0.67	0.65	0.68
	0.75	0.70	0.68	0.68	0.66	0.65	0.65	0.67	0.69	0.70	0.65	0.64	0.69	0.69	0.70	
OT2	0.65	0.72	0.67	0.68	0.66	0.72	0.73	0.75	0.72	0.68	0.67	0.71	0.74	0.70	0.73	0.70
	0.74	0.72	0.73	0.74	0.69	0.67	0.66	0.75	0.70	0.69	0.68	0.74	0.70	0.67	0.70	

Table VIII shows a reasonable result from a Pareto front with the same average hypervolume, given in Table VII, for each respective method. For example, row 1, **M**, is a result from a Pareto front with a hypervolume value of 0.67. Row 5, **M**(%) shows the percentage improvement of each objective over the original results shown in Table VI. We can then compare the average improvement of all objectives for each method. If we compare the average hypervolumes from Table VII to the average percentage improvement, we see a similar pattern. The standard NSGA-II (**M**) improves over the original results, though we can improve on each objective further with the fuzzy methods type-1 (T1) and type-2 (T2). There is no significant difference between the unoptimised fuzzy systems, but the optimised type-2 fuzzy systems (OT2) outperform all other methods. This method gives a 21.06% improvement in Coverage (C) a 3.55% improvement in travel and a 66.00% improvement in area balancing (AB).

TABLE VIII. EXAMPLE RESULTS FROM THREE-OBJECTIVE PROBLEM

	C (%)	T(km)	AB (hrs)	Average
M	89.07	23.70	249.21	-
T1	91.48	25.77	146.81	-
T2	91.94	25.12	150.88	-
OT2	92.15	25.56	120.58	-
M (%)	17.01	10.57	29.73	19.10
T1 (%)	20.18	2.75	58.60	27.18
T2 (%)	20.78	5.21	57.46	27.82
OT2 (%)	21.06	3.55	66.00	30.20

Also, we have shown that we can improve our modified NSGA-II (MOGA) even further by including the type-2 FLSs and pre-optimising the membership functions and footprints of uncertainty before the primary work area optimisation takes place. To conclusively say this is the case, we can perform statistical analysis on the two sets of hypervolume values given by the MOGA and the MOGA with optimised type-2 FLSs.

The P value given if we compare these two sets of hypervolume values, using Kruskal-Wallis is **0.0016**, or 0.16% significantly below the alpha value of 0.05 (or 5%) to show a statistically significant difference between the sets.

A. Qualitative Analysis

We can subjectively compare the results produced by each variation of the optimisation. Fig. 14 shows the original WAs. The WA marked as ‘1’ in Fig. 14 is a large city, and because the city is all in one WA, this is the cause of the large work imbalance that is shown in Table VI.

Fig. 15 shows a ‘best’ Pareto solution from the T1 variation which has identified the imbalance issue but only split the city up into 2 WAs. Given the scale of the imbalance, three city WAs might address the issue better. Fig. 16 shows us the T2 result. It generates the needed three city WAs, however area ‘3’ covers a large area to the south-east, this would cause travel time issues, and hence the travel objective has not been effectively captured as well as it could be.

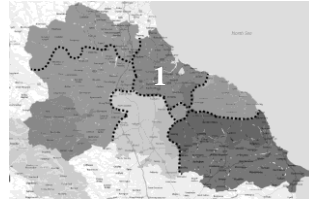


Fig. 14. Original WA Design Fig.



Fig. 15. A T1 Solution



Fig. 16. A T2 Solution



Fig. 17. An OT2 Solution

Fig. 17 shows an OT2 result. The city is split into three equal WAs (1-3) with the rural WAs outside and much larger. Giving us an operationally optimal solution.

VII. RESULTS FOR FUZZY DOMINANCE RULES FOR MANY-OBJECTIVE PROBLEMS

Our first set of experiments described in Section VI demonstrates that the use of optimised type-2 FLS improves the results for our multi-objective problem. However, we detailed we have five objectives in total, making this a many-objective problem. We discussed the issues surrounding parent selection for many-objective problems in IV D. We discussed that we believed it to be a problem with the crisp value comparison in the dominance rules. Hence, we present our results for our experiments using Fuzzy Dominance Rules (FDRs) described in Section 4 D. We will use a 10% grace value of the objective values when we calculate the dominance. As we are using five objectives, we cannot compare the hypervolume values from Table VII. We will compare our modified NSGA-II, the MOGA, with our modified NSGA-II with Fuzzy Dominance Rules (MOGA-FDR). To tie the whole system together, we will use our genetically optimised type-2 system with FDR to get the best improvement for our many-objective problem.

Now we are using more objective we have more reference points for the hypervolume. Once again, we multiply our maximisation objective by -1 when calculating the hypervolume. Our reference points are now 0, 100, 0, 850 and 150 for coverage, travel, utilisation, area balancing and team balancing respectively.

Table XI shows the hypervolume values for the 30 runs of each method, runs 1-15 on the first row and runs 16-30 on the second row of each method. For the MOGA, given by **M**. The average of these is 0.48. **FDR** gives the hypervolume values for the MOGA with FDRs implemented, with an average of hypervolume of 0.51. If we perform Kruskal-Wallis statistical analysis on these hypervolume sets, we get a P value of **0.049**, which is less than the required alpha value of 0.05 to show the difference in results is statistically significant. So far, we have shown that using genetically optimised type-2 systems and or the introduction of FDR statistically improves the Pareto front results independently. The final step is to measure the impact combining these two methods of improvement together.

TABLE IX. FDR HYPERVOLUME SUMMARY TABLE

																	Avg.
M	0.41	0.41	0.42	0.43	0.44	0.44	0.45	0.45	0.46	0.46	0.46	0.47	0.47	0.47	0.47	0.47	0.48
	0.47	0.47	0.48	0.49	0.49	0.49	0.49	0.51	0.52	0.53	0.53	0.54	0.56	0.57	0.60		
FDR	0.42	0.43	0.43	0.44	0.45	0.45	0.47	0.47	0.47	0.47	0.48	0.48	0.49	0.50	0.51	0.51	0.51
	0.51	0.51	0.52	0.52	0.52	0.54	0.54	0.55	0.56	0.57	0.57	0.57	0.57	0.59	0.63		
OT2	0.54	0.54	0.54	0.55	0.56	0.57	0.58	0.60	0.60	0.61	0.61	0.62	0.62	0.63	0.63	0.63	0.63
FDR	0.64	0.64	0.65	0.65	0.66	0.66	0.67	0.67	0.67	0.69	0.69	0.70	0.70	0.73	0.73		

The results for this are shown in Table IX as the **OT2FDR** hypervolume set. From the average hypervolume values, we can see OT2FDR has improved the average hypervolume by 24.29% if we compare it to the FDR results. If we perform the Kruskal-Wallis test we get a P value of **4.47⁻⁹** when comparing FDR to OT2FDR, and for completeness, we also get a P value of **1.86⁻¹⁰** if we compare our MOGA to OT2FDR. The winning method for each seed value is highlighted in bold, where the OT2FDR method wins in all cases.

Table X shows a reasonable result from a Pareto front with the same average hypervolume, given in table IX (similar to Table VIII). For example, row 1, M, is a result from a Pareto front, using the standard NSGA-II algorithm, with a hypervolume value of 0.48. Row 5, M(%) shows the percentage improvement of each objective over the original results shown in Table IX. From Table X we can see that the standard NSGA-II algorithm completely failed in the area balancing objective (AB) (rows 1 and 4). This outcome is expected given the problems we have described with multi-objective algorithms trying to handle more than three objectives. As a result, the average objective improvement is just 2.23%.

Once we add the fuzzy dominance rules, all objectives are handled much better. Though there is a small degradation in the travel objective (T) this can be explained by the 20.24% increase in the amount of covered jobs (C). Given that these are directly conflicting objectives (to complete more jobs, one must travel to the job) this is a good outcome. Finally, when we add the optimised type-2 fuzzy systems to the process, alongside the fuzzy dominance rules, all objectives are improved over the original results, with an average objective improvement of 30.85%. This is approximately the same level of improvement given in the three-objective problem results, shown in Table VIII. Thus, the system results in a consistent level of improvement, regardless of the number of objectives.

TABLE X. EXAMPLE RESULTS FROM FIVE-OBJECTIVE PROBLEM

	C (%)	T (km)	U (%)	AB (hrs)	TB (people)	Avg.
M	89.37	22.5	85.66	536.41	68.00	-
FDR	91.53	27.43	85.70	276.73	44.00	-
OT2FDR	92.27	24.53	86.40	189.17	34.00	-
M (%)	17.41	15.09	25.69	-51.25	4.23	2.23
FDR (%)	20.24	-3.49	25.76	21.97	38.03	20.50
OT2FDR (%)	21.22	7.44	26.78	46.66	52.11	30.84

These results make a strong case for both types of fuzzy system to be introduced in our optimisation algorithm.

VIII. REAL WORLD IMPACT OF IPATCH

The developed iPatch tool (shown in Fig. 18) was implemented with the goal of improving the organisational design of a mobile workforce within BT. The work presented in this paper looks at the geographical optimisation, which is significantly more developed than the resource optimisation functionality. Published work on resource optimisation can be found in a British Computer Society (BCS) award-winning paper [2].

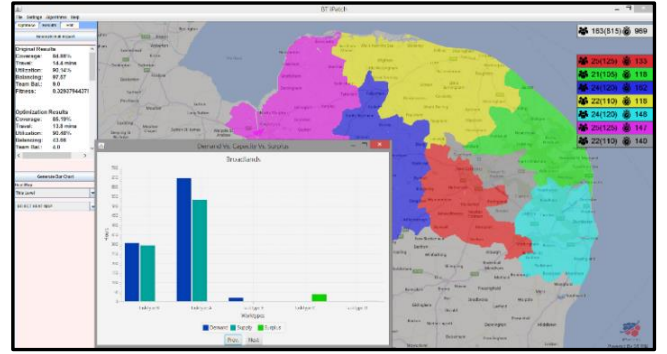


Fig. 18. BT's iPatch Tool

The application was developed in close collaboration with the users of the system. This, in turn, allowed detailed feedback on problems that came to light throughout development, which allowed the results produced by iPatch to translate into the real-world successfully.

iPatch has generated an increase in productivity of 0.5% saving an estimated £1 million a year over the first two years. It also cut fuel consumption by 2.9%, an additional saving of over £200K a year. In addition to the financial benefits, customer commitments are more effectively met, improving the service quality, and due to less fuel consumption, the company can promote sustainability targets with less CO₂ emitted. Over the period of deployment, iPatch has reduced CO₂ emissions by more than 2500 metric tonnes.

Furthermore, a report by the UK's Department of Transport found that for every billion vehicle miles travelled there were 15,409 serious injuries or deaths, or 1 per 64,900 miles [27].

As we have saved an estimated 7.7 million miles of travelling, this equates to potentially saving 118 casualties and fatalities. The system won the 2015 Global Telecommunications Business award for the best business innovation of the year in its first year of use [28] and was highly commended at the IET Innovation Awards 2016 [29].

These outcomes show the real world impact these AI technologies, including advanced fuzzy logic systems, are having on a large, nationwide, mobile engineering workforce.

IX. CONCLUSION & FUTURE WORK

In this paper, we have presented our work on iPatch, a many-objective fuzzy logic system for the optimisation of a field workforce. This system includes a distance metric for analysing the solutions that are generated by a multi-objective optimisation algorithm and to help with parent selection. We implemented a type-1 and type-2 FLSs to improve our applications results. We then showed that genetically optimising the type-2 FLSs gave us a real improvement when comparing the hypervolumes of the MOGA and the genetically optimised type-2 FLSs. The P value here was 0.0016 significantly below the required 0.05 to show statistical significance.

We then extended our work by looking at how we could solve the many-objective issues given by standard crisp dominance rules. We again showed that by including FDRs in our MOGA we improved on the hypervolumes given by the Pareto fronts. The P value attained here was 0.048, again lower than the required 0.05. Combining the fuzzy systems and FDRs resulted in a significant improvement to the many-objective algorithm, with a P value of 1.8647^{-10} when compared to the standard MOGA we previously used.

The fuzzy approaches, particularly the optimised type-2 components of the system, result in an average level of objective improvement of approximately 30%, regardless of the number of objectives.

Over the first two years of deployment by British Telecom, this application has *increased productivity by 0.5% across the mobile workforce and reduced fuel consumption by 2.9%. This has led to a productivity benefit of £1million a year and a saving of over £200K a year in fuel costs.*

The secondary benefits have also been measured. As the engineers are travelling less this has *saved an estimated 2,500 metric Tonnes of CO₂ and potentially reduced serious traffic casualties or fatalities by over 100.*

For our future work, we intend to explore the impact of type-2 FDRs and to expand the workforce optimisation capabilities of iPatch.

X. REFERENCES

[1] N. Safaei, D. Banjevic, A. K. S. Jardine ‘Workforce Planning for Power Restoration: An Integrated Simulation-Optimization Approach’ IEEE Transactions on Power Systems Vol. 27 (1) 2012 pp. 442 - 449

[2] A. Mulla, G. Raravi, T. Rajasubramaniam, R.P. J. Chandra Bose, K. Dasgupta ‘Efficient Task Allocation in Services Delivery Organizations’ IEEE International Conference on Services Computing (SCC), San Francisco, USA, 2016, pp.555 – 562

[3] R. Baraglia, J.I. Hidalgo, R. Perego “A hybrid heuristic for the traveling salesman problem” IEEE Transactions on Evolutionary Computation, Vol. 5, No. 6, pp. 613-622. 2001.

[4] D. Soria, J. M. Garibaldi, AR. Green, D. G. Powe, C. C. Nolan, C. Lemetre, G. R. Ball, I. O. Ellis “A Quantifier-Based Fuzzy Classification System for Breast Cancer Patients” Artificial intelligence in medicine Vol 38 (3), 2013, pp.175-184

[5] M. Zouari, S. Cherif, H. Kammoun, H. Lajmi, A. M. Alimi ‘Towards type-2 fuzzy rule base system for road choice’ International Conference on Intelligent Systems Design and Applications (ISDA), Marrakesh, Morocco, 2015 pp. 243 – 248

[6] S. Miller, V.Popova, R. John, M. Gongora, Improving resource planning with soft computing techniques, Proceedings of the 2008 UK Workshop on Computational Intelligence, Leicester, UK pp.37-42, 2008

[7] M. Cimitile, M. Gaeta, V. Loia ‘An ontological multi-criteria optimization system for Workforce Management’ IEEE Congress on Evolutionary Computation (CEC), Brisbane, Australia, 2012 pp.1-7

[8] A.Starkey, H.Hagras, S.Shakya, G.Owusu, “A Genetic Algorithm Based Approach for the Simultaneous Optimisation of Workforce Skill Sets and Team Allocation”, AI-2016, Cambridge United Kingdom, pp. 253-266, 2016

[9] H.Ishibuchi, N. Tsukamoto, Y. Nojima “Evolutionary many-objective optimization: A short review”, IEEE Congress on Evolutionary Computation, Hong Kong, pp. 2419 – 2426, 2008

[10] Y.Liu, D. Gong, J.Sun, Y.Jin, “A Many-Objective Evolutionary Algorithm Using A One-by-One Selection Strategy”, IEEE Transactions on Cybernetics, pp. 2168-2267, 2016

[11] H. Algethami, R. Lankaites Pinheiro, D. Landa-Silva “A genetic algorithm for a workforce scheduling and routing problem” IEEE Congress on Evolutionary Computation, Vancouver, Canada, pp. 927-934, 2016

[12] N. K. Sharma ; R. M. R. Guddeti “Multi-Objective Resources Allocation Using Improved Genetic Algorithm at Cloud Data Center”, IEEE International Conference on Cloud Computing in Emerging Markets, Bangalore, India, pp. 73 – 77, 2016

[13] T. Yamada, R. Nakano ‘A genetic algorithm with multi-step crossover for job-shop scheduling problems’ First International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications, Sheffield, UK. 1995. Pp.146 - 151

[14] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, “A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II”, IEEE Transactions on Evolutionary Computation, Vol. 6, No. 2, pp. 182-197. 2002.

[15] A. Ibrahim, S. Rahnemayan, M. Vargas Martin, K. Deb, “3D-RadVis: Visualization of Pareto front in many-objective optimization, IEEE Congress on Evolutionary Computation, Vancouver, Canada pp 736 – 745, 2016

[16] E. Zitzler, L. Thiele, M. Laumanns, C.M. Fonseca, V.G. da Fonseca, “Performance assessment of multiobjective optimizers: an analysis and review” IEEE Transactions on Evolutionary Computation Vol.7 No. 2 pp117-132, 2003

[17] A. L. Custódio, M Emmerich, J. F. A. Madeira, “Recent Developments in Derivative-Free Multiobjective Optimisation” Computational Technology Reviews Vol. 5 pp.1 – 31 2012.

[18] N. N. Karnik, J. M. Mendel, “Centroid of a Type-2 Fuzzy Set”, Information Science Vol 131, pp. 195-220, 2001

[19] H. Hagras, “A Hierarchical Type-2 Fuzzy Logic Control Architecture for Autonomous Mobile Robots”, IEEE Transactions on Fuzzy Systems, Vol. 12, No. 4, pp. 524-539, 2004.

[20] C. Lynch, H. Hagras and V. Callaghan, “Embedded Type-2 FLC for Real-Time Speed Control of Marine & Traction Diesel Engines” IEEE International Conference on Fuzzy Systems. Reno, NV, USA, pp. 346-352, 2005.

[21] H. Hagras, “Embedding Computational Intelligence in Pervasive Spaces”, IEEE Pervasive Computing, vol.6, no.3, pp. 85-89, May/June 2007.

[22] M. Raoufi, N. G. Seresht, A. Robinson Fayek ‘Overview of fuzzy simulation techniques in construction engineering and management’ Fuzzy Information Processing Society (NAFIPS), 2016 Annual Conference of the North American, El Paso, Texas, USA, 2016 pp.1-6

[23] A. Starkey, H. Hagras, S. Shakya, G. Owusu “A multi-objective genetic type-2 fuzzy logic based system for mobile field workforce area optimization” Information Sciences Vol. 329 p.390-411, 2016

[24] A. Starkey, H. Hagras, S. Shakya, G. Owusu, “Fuzzy Dominance Rules for Real-World Many Objective Optimization”, IEEE International Conference on Fuzzy Systems, Naples, Italy, July 2017

[25] C. Wagner, H. Hagras “A Genetic Algorithm Based Architecture for Evolving Type-2 Fuzzy Logic Controllers for Real World Autonomous Mobile Robots” IEEE International Conference on Fuzzy Systems, London, United Kingdom, pp. 193 – 198, 2007

[26] A. Starkey, H. Hagras, S. Shakya and G. Owusu “A Many-Objective Genetic Type-2 Fuzzy Logic System for the Optimal Allocation of Mobile Field Engineers”, IEEE International Conference on Fuzzy Systems, Vancouver, Canada, pp. 2051 – 2058, 2016

[27] Department for Transport, 25th June, 2015: www.gov.uk/government/uploads/system/uploads/attachment_data/file/46304/5/rcgb2014-01.pdf [Last Accessed: 13th April 2017]

[28] University of Essex News, School of CSEE: www.essex.ac.uk/csee/news_and_seminars/newsEvent.aspx?e_id=7696 [Last Accessed: 13th April 2017]

[29] University of Essex News, School of CSEE: www.essex.ac.uk/csee/news_and_seminars/newsEvent.aspx?e_id=11882 [Last Accessed: 13th April 2017]