# A Decision Support System for a Real-Time Field Service Engineer Scheduling Problem with Emergencies and Collaborations

*Hanna Grzybowska[1], Charles Gretton[2,3], Philip Kilby[2] and S. Travis Waller[1,2]*

*[1]School of Civil and Environmental Engineering, UNSW Australia*
*[2]NICTA*
*[3]Griffith University, Gold Coast*

## Abstract

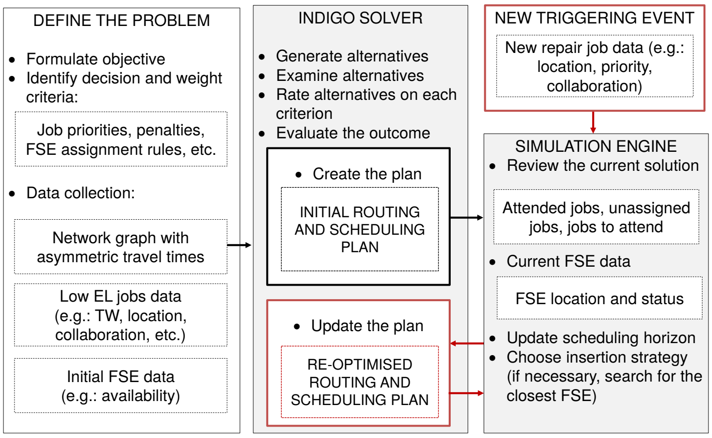
We treat a dynamic routing and scheduling problem by repeatedly re-planning using a heuristic for solving a variety of the Vehicle Routing Problem. The problem we treat occurs in the situation when Field Service Engineers are assigned a sequence of jobs to attend. The jobs are geographically distributed and not all jobs to be undertaken are known in advance of planning. This dynamic occurrence of job requests is stochastic. Jobs are assigned an Emergency Level, which is the highest for the repair jobs involving a person in danger. In addition some jobs require two engineers to attend. We refer to such jobs as collaborative. Our approach re-schedules the pending jobs in an event-driven manner. The event-driven scheduling process ensures that jobs of high importance, with a high emergency level, are completed promptly. The proposed Decision Support System assists in the decision making concerning the management of Field Service Engineers, in the case when real-time information is available. Its architecture includes two main modules: Simulation Engine and Indigo Solver, using a flexible heuristic based on an Adaptive Large Neighborhood Search. We find that our approach of event-driven re-planning is able to plan for real-world scenarios using significantly fewer resources than are employed in practice.

***Index Terms** – Field Service Engineer; Staff Scheduling; Dynamic Vehicle Routing Problem; Simulation; Decision Support System*

## Real-Time Field Service Engineer Scheduling Problem with Emergencies and Collaborations

Given a set of known jobs, determine for the smallest possible number of FSEs, a set of routes with a corresponding schedule, so that: each non-attended job location is visited exactly once; each route starts and ends at an FSE's residence location; an FSE attends one and only one job at a time; each job is attended within a specified hard TW; each job is attended within FSE's work hours; collaboration constraint is respected; the collaborating FSEs always finish attending the job together; the cost of the first and last trip on a route is not included in the entire routing cost; entire routing cost is minimized.

## Framework of Proposed Decision Support System



DEFINE THE PROBLEM

- Formulate objective
- Identify decision and weight criteria:

Job priorities, penalties, FSE assignment rules, etc.

- Data collection:

Network graph with asymmetric travel times

Low EL jobs data (e.g.: TW, location, collaboration, etc.)

Initial FSE data (e.g.: availability)

INDIGO SOLVER

- Generate alternatives
- Examine alternatives
- Rate alternatives on each criterion
- Evaluate the outcome

- Create the plan

INITIAL ROUTING AND SCHEDULING PLAN

- Update the plan

RE-OPTIMISED ROUTING AND SCHEDULING PLAN

NEW TRIGGERING EVENT

New repair job data (e.g.: location, priority, collaboration)

SIMULATION ENGINE

- Review the current solution

Attended jobs, unassigned jobs, jobs to attend

- Current FSE data

FSE location and status

- Update scheduling horizon
- Choose insertion strategy (if necessary, search for the closest FSE)

All static route planning and scheduling uses a flexible heuristic based on an Adaptive Large Neighborhood Search implemented in Indigo

## Problem Definition

$$min\sum_{t\in T}\sum_{k\in K}\sum_{(i,j)\in A}c_{ij}x_{ijkt}+\sum_{i\in N}w_i$$

Subject to:

$$\sum_{t\in T}\sum_{k\in K}\sum_{j\in N}x_{ijkt}=1 \qquad \forall i\in N$$

$$\sum_{i\in N}x_{ijkt}-\sum_{i\in N}x_{jikt}=0 \qquad \forall j\in N,k\in K,t\in T$$

$$\sum_{i}\sum_{j\neq i}x_{jikt}\leq|S|-1 \qquad \forall S\in N_c,|S|\geq 2\ k\in K,t\in T$$

$$x_{ijkt}\left(z_i+s_i+c_{ij}-z_j\right)\leq 0 \qquad \forall i\in N,j\in N,k\in K,t\in T$$

$$E_H\leq E_{kt}\leq e_i\leq z_i\leq l_i \qquad \forall i\in N,k\in K,t\in T$$

$$z_i+s_i\leq l_i\leq L_{kt}\leq L_H \qquad \forall i\in N,k\in K,t\in T$$

$$w_i=max\{0,e_i-a_i\} \qquad \forall i\in N$$

$$z_i=max\{a_i+w_i,0\}$$

$$z_i+s_i=z_{p(i)}+s_{p(i)} \qquad \forall i\in N,p(i)\in N$$

$$w_i\geq 0\ \forall i\in N$$

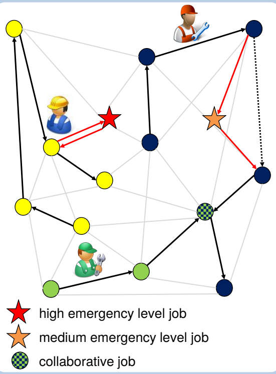
$$z_i\geq 0\ \forall i\in N$$

$$a_i\geq 0\ \forall i\in N$$

$$x_{ijkt}\in\{0,1\} \qquad \forall i\in N,j\in N,k\in K,t\in T$$

## Notation

- $G=(N,A)$ complete graph, where:
  - ✓ $N=\{0,1,\dots,n\}$ set of jobs
  - ✓ $A=\{(i,j): i,j\in N, i\neq j\}$ set of arcs
- $c_{ij}$ - nonnegative cost associated with arcs $c_{ij}\neq c_{ji}\ \forall i,j\in N$
- $[E_H, L_H]$ - bounded scheduling horizon consisting of $T$ work-days
- $[E_{kt}, L_{kt}]$ - hard availability TW of FSE $k\in K$ at particular work-day $t\in T$
- $[e_i, l_i]$ - hard TW of job $i$
- $s_i$ - service time duration of job $i$
- $a_i$ - arrival time at the job $i$
- $w_i$ - waiting time at the job $i$
- $z_i$ – service start time at the job $i$
- $p(i)$ - collaboration part of job $i$
- $\tau_j$ - time instant when a new repair job $j$ is called-in



## Experiment Design

We consider three main experimental design levels:

- number of repair jobs
- emergency levels of repair jobs
- collaborations - when a job needs to be attended in parallel by two FSEs

| | Maintenance jobs (low EL) | | | Repair jobs (medium EL) | | | Repair jobs (high EL) | | |
|---|---|---|---|---|---|---|---|---|---|
| **Sce** | **#** | **Collab%** | **Duration** | **#** | **Collab%** | **Duration** | **#** | **Collab%** | **Duration** |
| **1** | 420 | 0% | EXP(109.5) | 0 | 0% | - | 0 | 0% | - |
| **2** | 620 | 0% | EXP(109.5)<br>EXP(38.1) | 0 | 0% | - | 0 | 0% | - |
| **3** | 420 | 0% | EXP(109.5) | 200 | 0% | EXP(38.1) | 0 | 0% | - |
| **4** | 420 | 0% | EXP(109.5) | 180 | 0% | EXP(38.5) | 20 | 0% | EXP(41.9) |
| **5** | 420 | 5% | EXP(109.5) | 0 | 0% | - | 0 | 0% | - |
| **6** | 620 | 5% | EXP(109.5)<br>EXP(38.1) | 0 | 0% | - | 0 | 0% | - |
| **7** | 420 | 5% | EXP(109.5) | 200 | 5% | EXP(38.1) | 0 | 0% | - |
| **8** | 420 | 5% | EXP(109.5) | 180 | 5% | EXP(38.5) | 20 | 5% | EXP(41.9) |

- ❖ number of FSEs is constant and equal to 10
- ❖ scheduling horizon is 20 work-days (8 hours long)
- ❖ half of the FSEs start work at 7am and half starts at 9am
- ❖ job duration was modelled using exponential distribution:
  - ✓ $\lambda$=0.00913 for jobs with low EL (420 jobs)
  - ✓ $\lambda$=0.026 for jobs with medium and high EL (200 jobs)
- ❖ length of time intervals between call-in times of repair jobs was modelled using exponential distribution:
  - ✓ scenarios containing only repair jobs with medium EL $\lambda$=0.00547
  - ✓ scenarios containing both medium and high EL repair jobs:
    - ➢ $\lambda$=5.4532E-4 for jobs with high EL
    - ➢ $\lambda$=0.00492 for jobs with medium EL
- ❖ collaborations correspond to 5% of the total jobs number
- ❖ travel time and distance are real-life values plus noise

## Policy

The policy defines additional constraints regarding job's priority which reflect the *emergency level*:

- *high* – repair jobs to be performed immediately
- *medium* – repair jobs to be performed the same day the machinery failed
- *low* – maintenance jobs

- ❖ repair jobs have priority over maintenance jobs
- ❖ maintenance jobs can be suspended
- ❖ suspended job always has to be finished by the same FSE who started it

## Experiment Results

- The Simulation Engine was developed using Python 2.7.3 [MSC v.1500 64 bit (AMD64)] on win32
- The Indigo Solver was implemented in C++

| | | #Attended jobs | | Total Solution Evaluation | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Sce.** | **# routes** | **Maintenance (low EL)** | **Repair (high/medium EL)** | **Travel time (min)** | **Waiting time (min)** | **Service time (min)** | **Solution cost (min)** | **Total CPU (sec)** | **Avg CPU (sec)** |
| **1** | 200 | 420.00 | 0 | 17.30 | 0.00 | 53383.00 | 53400.30 | 32.90 | 32.90 |
| **2** | 200 | 620.00 | 0 | 51.50 | 0.00 | 58802.00 | 58853.50 | 47.46 | 47.46 |
| **3** | 200 | 420.00 | 200 | 785.10 | 16158.70 | 60919.00 | 77862.80 | 1052.74 | 5.24 |
| **4** | 200 | 419.59 | 200 | 1608.00 | 17295.90 | 60762.20 | 79666.10 | 954.41 | 4.32 |
| **5** | 200 | 420.00 | 0 | 107.10 | 20.40 | 53383.00 | 53510.50 | 168.46 | 168.46 |
| **6** | 200 | 620.00 | 0 | 216.80 | 8.90 | 58618.80 | 58844.50 | 249.04 | 249.04 |
| **7** | 200 | 418.90 | 200 | 2223.50 | 16891.20 | 60734.30 | 79849.00 | 3555.83 | 17.69 |
| **8** | 200 | 417.24 | 200 | 2959.10 | 18362.00 | 60659.10 | 81980.20 | 3767.87 | 17.05 |

- ❖ all the high and medium EL jobs were attended in all the scenarios
- ❖ travel time values are the highest in the scenarios considering dynamic requests
- ❖ waiting time appears only in the scenarios considering dynamic requests (~20 per cent of the total solution cost)
- ❖ consideration of collaborative jobs has big impact on the travel time values

## Summary

In sum, we define and computationally explore on a basis of simulation a methodological proposal for a DSS to assist in the decision making concerning the management of FSEs, when real-time information is available. We provide a model for the Real-Time FSEEC problem trying to carefully emulate the real-life case. We consider collaborative jobs and define different emergency levels. Our approach rebuilds the current routing plan using Indigo Solver and defines the policy treating the repair jobs in accordance with their emergency levels. We show the big impact that dynamic requests have on the solution and the benefits which might be provided by the event-driven re-planning and re-scheduling. Our reactive re-planning approach is able to schedule engineers for all emergencies and satisfies the vast majority of periodic maintenance tasks. In the tested synthetic scenarios all the dynamic requests were attended over all runs. The work was inspired by a commercial engagement and we have found that our approach is able to plan for real-world scenarios using significantly fewer resources than are employed in practice.