Collaborative Evolutionary Multi-Project Resource Scheduling

1. Introduction

Many real world scheduling problems, particularly those which arise in the large-scale construction, corporation wide logistics, or large-scale software development, are *multi-project resource constrained problems*, with a particular emphasis on the word *multi*. Such problems typically involve many thousands or tens of thousands of tasks, grouped into several hundred or more sub-projects. The individual sub-projects are tightly linked by project-wide resources and other logistic interdependencies. Despite the large-scale complexity, a feature of such problems is usually their high level management by a single individual. This individual, who we will refer to as the *scheduler*, maintains a project-wide view and attempts to organise sub-projects relative to each other in time in such a way that multiple criteria are satisfied, many of which cannot be easily formalised. For example, although it may be possible to schedule a certain sub-project early within the overall project schedule, and entirely within a two-month period, experience may tell the scheduler to move this sub-project into the middle of the overall plan, spread out over three months. This might mean that hired resources utilised by this sub-project which are also used by sub-projects in the middle of the overall plan could be used far more cost-effectively and efficiently used; the scheduler may feel that some slippage in the sub-project's desirable due date can be reasonably traded off against this benefit. Many similar, and many quite different motives may underpin the scheduler's development of the overall project plan.

There remains a general need to complete the entire project to a reasonable deadline and fulfil a combination of makespan and just-in-time oriented goals for the individual subproject. However, the additional concerns of shared resources combined with the scheduler's general knowledge base of barely or non-formalisable rules makes the overall multi-project resource-constrained scheduling problem particularly hard for the academic scheduling research community. The reasons for this are well known for being problematic aspects of real-world scheduling, but seem exacerbated in the massive multi-project resource-constrained scenario. For example, there is a constant need for rescheduling in multi-project scenarios, as plans alter for various reasons, certain sub-projects finish unexpectedly early or late, and so forth. This seems to undermine the utility of systems which perform particularly well on, say, a pure job-shop style sub-project with guaranteed resources, since it is unlikely that the derived optimal or near-optimal schedule for this sub-project will be applicable when it is due to start.

Arguably more problematic is the multi-criteria natire of multi-project scenarios. Indvidual sub-projects may have different sets of criteria, while the overall project-wide goals of the scheduler may at least in part be difficult or impossible to formalise.

In this paper we propose and explore a way to deal with multi-project resource-constrained scheduling problems, which is designed to combine standard automated scheduling with scheduler guidance. A collaborative evolutionary multi-project scheduler is described. It uses an evolutionary algorithm to evolve schedules according partly to standard criteria of due date slippage and makespan, but also includes the scheduler's input as part of a schedule's selective fitness. The scheduler regularly views example GANTT charts from the current population, and assigns coarse relative fitnesses to them (via an easy-to-use interface) by considering the overall 'shape' of the developing schedules. Results so far, as well as potential user comments, are very promising for the method.

The remainder of this paper is set out as follows. In Section 2 we describe the results of a survey which tested the idea of a 'collaborative evolutionary scheduler' on a collection of real industry schedulers who currently use a mixture of techniques ranging from commercial automated scheduling packages specialised to their domain, through to simple 'linear scheduling' software, or in many cases pen and paper. The survey unequivocally indicated that such collaborative scheduling software would be welcomed in industry. In section 3 we briefly review collaborative

1.1 Multi Project Resource Scheduling

This paper explores a potential solution to the problem of Scheduling the Resources required by a significant number of independent Project Plans within an organisation, subject to a range of constraints including minimising the disturbance to the existing scheduled dates of the individual plans and the efficient use of common resource pools. The overall system of multiple projects scheduled together is referred to as a Programme.

The essential principle of this paper is that scheduling a Programme made up from the existing schedules of individual Projects is more than just a "Big Project Scheduling Problem".

Existing commercial systems would merge all the project data into one 'Big Project', and schedule the amalgamated project tasks in one pass using a Serial Scheduler. Also note that a Serial Scheduler only takes into account explicit information defined within the individual project plans.

1.2 Project Manager's "Gut Feel"

The project plans have explicit information for each activity in the form of start and end dates, durations, and logical interconnections which will have been detailed by the Project Manager using a computer based system. There may also be a priority order for the sequence of scheduling the projects.

These plans have other items of information that the Project Manager will implicitly know, but have no means of specifying explicitly within the software package. This information may be Fuzzy, reflecting the Project Manger's experience and 'gut feel'.

In particular the layout, arrangement and logic of the plans will reflect the Project Manager's understanding of a wide range of factors outside of the basic time, resource and cost parameters. These may include knowledge of suitability of co-workers (the effect of using several resources together), climate and time of year factors (the effect of winter for construction projects, or the effect of tides on oceanic oil exploration), and even cultural information (religious or other factors in various parts of the world).

A most important requirement for any Programme Scheduling System is that it should be able utilise as much of this implicit information as possible, such that the Project Manager of each project plan is not "surprised" by radical changes in the dates or logic of his original plan as a result of the Programme Scheduling process.

However, it is difficult (if not impossible) for the Project Manager, or indeed the software package, to specify this 'Fuzzy' information in a simple algorithmic fashion. The system needs to allow the Project Manager a mechanism for using his experience and gut feel to influence the solution finding process, but in an intuitive, fuzzy, human interactive manner.

1.3 Outline of the paper

First the paper explores what these implicit factors might be, by reviewing the results of a survey carried out with a series of Project Managers, from a range of industries.

Next the paper looks at the available techniques for scheduling projects and resources, and then discusses ways of enabling the Project Manager to use his implicit information in assisting the solution process.

The third section provides an overview of the design of the interactive interface, and explains how the Project Manager is able to influence the solution process.

Finally the paper provides results of experiments which show how the Project Manager is able to influence and drive the solution, and concludes with some ideas for further learning and adaption capabilities.

2. Survey

To determine what implicit factors are known, and possibly used, by Project Managers, a survey has been carried out with a number of Project Managers in a range of industries. The industries included organisations involved in Aerospace, Defence, Construction, Manufacturing, Government and Services.

The following table summaries the various issues and factors derived from the survey:

Many Informal Project interdependencies, not explicit in project plan
Formal Project/Task Links between different projects, not supported by software.
Many Unrelated Work Packages, e.g. Service Visit scheduling
Urgency of Delivery Dates (i.e. End Date important)
Customer Satisfaction Issues (kept informed of expected end dates)
Environmental Issues
Time of Year
Seasonal Availability of Resource (Holidays, Student availability)
Weather Patterns (e.g. Oil Rig completion in Spring or Summer)
Political Issues
Geographical Issues
Optimising Key Resource Usage
Resource Skills
Conflicting Project Priorities

As can be seen from the list, some are measurable, some subjective. In most cases there were conflicts between the various implicit issues, which would be resolved in an ad-hoc fashion by the Project Manager.

3. Solution Design

The requirement is to enable a Project Manager to set up the explicit information for a range of project plans, using a standard software package. The system should then attempt to find acceptable solutions using the implicit information from the Project Manager, in an intuitive, interactive fashion.

A simple Serial Scheduler will provide a one-off solution, which will either be acceptable or not. However this does not give the Project Manager any opportunity for interacting with the solution. He can merely reset the priority order of the schedule and try again.

It would be preferable to provide the Project Manager with a range of solutions for him to choose from, and for the system to be able to learn from the user's selection and therefore generate better solutions in a subsequent scheduling process.

A suitable method for generating multiple solutions is the Genetic Algorithm (GA), which effectively breeds a range of solutions. We then need some mechanism for displaying the solutions to the user, and picking up his selections and learning from them.

The graphical representations used by Project Managers include :

1. The Gantt (bar) chart, showing the start and end dates of the plans as horizontal bars against a timescale.
2. The Resource Profile, of resource usage against time.

It would be sensible to use these familiar representations as the GUI for the Project Manager so that the meaning of the results is clear and easy to understand.

The main feature of a GA is the Fitness Function, which is the algorithm used to determine the best (or fittest) solutions, which are then used to breed the next generation. In this interactive system the human user's selections of the best solutions will be used as inputs to the Fitness Function, and thus the next generation will breed new solutions derived from those which the Project Manager considers to be the best.

4. The Fitness Function
The GA Fitness Function is derived from one or more numeric attributes of the individuals in a given generation. These attributes must be measurable and relevant.

In the current literature the existence or ability to define a fitness function is often taken for granted, or assumed to be reasonably trivial to define.

- David Goldberg says "*Intuitively, we can think of the [fitness] function as some measure of profit, utility or goodness that we want to maximise*." [1].
- In [9] Salvatore Mangano explicitly assumes that the GA will have a calculable fitness function: *"…a crucial component of all GAs - the fitness function…is used to map the individual's bit strings into a positive number called the individual's fitness*".
- Melanie Mitchell says in [10] that "*The GA most often requires a fitness function that assigns a score (fitness) to each chromosome in the current population*".
- In [11] Darrell Whitley states "…*the evaluation* (fitness) *function is usually given as part of the problem description*".
- Stephanie Forrest in [12] considers the issues in defining the fitness function with "*In some cases the fitness function is obvious, but in many cases, it is not, and in all cases the particular fitness function selected will determine the ultimate success of the GA*."

None of these references follow up their discussion of the need for a fitness function with any details of the kinds of problems involved with the derivation of these fitness functions, and in particular whether the fitness function can be determined from the numerical parameters defining each member of the population.

For some applications the ability to measure numerical parameters is not simple, specifically where the fitness is related to the subjective opinions of the human user. If we look at applications where part (or all) of the fitness is a subjective assessment, then the question is not how to code the fitness function, but rather how to enable the human operator to specify his fitness parameters. Some examples of GA systems which require human interaction are discussed in section 3 below and include functions such as the generation and selection of Police Photofits [2] or the selection of auto-generated music [3] or computer "Art" [14-16].

In particular, this paper is interested in the problem of using GAs to perform Multi-Project Resource Scheduling, where a number of individual projects, using resources from a common pool, need to be scheduled as a whole rather than individually. In this problem the successful schedules are based on Fitness criteria relating to both **measurable parameters** (such as minimisation of project slippage, and minimisation of extension to project timescale) and also to **subjective parameters** based on the human operator's knowledge and experience (such as the 'rightness' of the shape of the new schedule, or knowledge of external factors affecting the allowable timeslots for projects that cannot be easily modelled within the schedule).

The requirement is to allow the human operator to enter his subjective choices at the point in the GA cycle where the fitness is calculated. This may replace the standard fitness function altogether, or be in addition to a basic fitness estimation. We need to allow the human being to be 'integrated' with the computer system in an efficient and straightforward manner.

The most important feature of a system that attempts to integrate the human operator within the Fitness function is its ability to let the operator review each of the potential solutions and to allow him to 'grade' their fitness using his intuition, experience and 'gut feel'. The state of the current population must be displayed to the operator, and probably the most appropriate medium is a picture, rather than pages of textual reports. In [13] Edward Tufte states "*Graphics reveal data. Indeed graphics can be more precise and revealing than conventional* [tabulations]".

We will now consider the best method for a computer based GA tool to interact with the human operator.

5. Human Interaction via the Graphical User Interface (GUI)
Computer systems lend themselves to graphical representation of information. Spreadsheets provide Histograms of data, project planning tools use the Gantt Chart to display the project plan, financial and accounting systems revel in Piecharts and XY charts. In most cases these

system provide ‘Read Only’ data, but some, such as the project planning tools, are interactive - allowing the user to change the data directly on the displayed picture.

The modern PC with its high-resolution monitor, wide range of colours and point-and-click interaction with the Mouse is a very suitable vehicle for the interaction between a computer program and a human being. Witness the sophistication of some of the current computer games which present complete 3D environments for the player to interact with.

5.1 Biomorphs

In “*The Blind Watchmaker*” [4] Richard Dawkins has used a PC program to simulate the evolution of creatures (‘Biomorphs’) which use numeric ‘genes’ to determine the growth of various tree-like forms. Dawkins displays the individuals in the current generation as pictures on the PC screen, and allows the human operator to select the parent for the next generation. He calls this ‘artificial selection’ in the same way that human breeders create new forms of Dogs or Pigeons, and the process of selecting the next parent is entirely subjective, based on the personal opinion and aesthetic sensibilities of the human user. In “*Climbing Mount Improbable*” [5], Dawkins continues his theme with the development of Spider’s webs, sea shells and arthromorphs (PC based ants, spiders and beetles).

5.2 Interactive AI scheduling

The technique of providing a GUI, for human interaction with the basic computer algorithm, has been used in the following examples, which demonstrate the possibilities of human interaction with scheduling systems. (Note that the following examples are not GA based, but use AI and Expert Systems methods.)

“*Constraint-Oriented Cooperative Scheduling for Aircraft Manufacturing*” [7] proposes a interactive human-machine system for solving scheduling problems in an aircraft manufacturing workshop. The system required the computer based scheduler to take into account **specified constraints** (such as due dates, processing times and machine capacities) as well as **subjective constraints** (such as operator preferences).

The authors state that “*rather than capturing the know-how of human decision-makers through a restrictive rule based model, we decided to integrate humans, rather than simulate them, into the solution process*”. The computer checks the consistency of thousands of constraints and the humans select the most appropriate choices from interactive computer displays of the potential solutions.

The authors summarise their report with “*this project demonstrates the validity of the idea that computers must support human decision-makers, rather than imitate or replace them*”.

“*Knowledge-Based Scheduling systems in Industry and Medicine*” [8] describes a computer-based scheduling system designed to enhance the capabilities of the human operators. It is based around a graphical user interface which displays the information about the status of the scheduled activities and supports interactive scheduling. The following features are highlighted:

- Comprehensive display of all relevant information on one screen.
- Visualisation of the current state of the problem solving process.
- User monitoring of the effects of scheduling decisions.
- Interactive creation, modification and repair of the schedule.

The system performs in three main stages:
1. Initial Schedule : a computer algorithm to determine the feasible schedules,
2. Human Interaction : to select or alter the schedules.
3. Checking : the feasibility of the human alterations.

The authors note “*the user should be able to monitor all scheduling actions*” and that “*industrial scheduling systems should support the interactive as well as automatic part of scheduling*”.

All the above systems take advantage of the power of modern PCs and the Windows environment to present all relevant data to the user in an easily accessible format - the graphical user interface. They then allow the user to interact with the system to provide the human experience, judgement and subjective preferences in a very flexible format. They balance the crude power of the computer to churn through the basic algorithms delivering a range of potential solutions with the lateral thinking of the human.

These techniques can be very useful in our search for the best way to embed the human operator's experience and intuition in a computer based GA system. We will now look at the potential for the Interactive GUI linked to a GA.

6. Interactive Interfaces to Genetic Algorithms

A review of the current literature (in particular the Handbook of Evolutionary Computing) to find existing systems that provide a human interface to the GA fitness function found the following packages:

6.1 Interactive GA PhotoFits

"*FacePrints*" [2] is package that uses subjective human recognition to guide a GA search through 'face space' of billions of possible facial composites. A witness interactively exerts influence over the selection process by using subjective estimates of the likeness of images to the remembered features of the culprit.

The system has been developed from the computerised PhotoFit techniques which use over 600 interchangeable photographs to build a composite image of the culprit, but relies on the ability of the witness to recall the various features of the face. The *FacePrints* package uses a GA and relies on the powerful human capacity for recognition rather than the more dubious memory based recall.

The system dynamically interacts with the witness, and allows a dynamic search for the culprit's face within the multi-dimensional face space. Each parameter of the GA describes specific values of the various features and proportions of the face which are encoded into the unique genotypes. The GA generates populations as normal and then displays graphically the 30 generated images. The witness enters ratings in the range '0' to '9' purely on the likeness of the image rather than any specific attribute. The GA then generates the next population using these ratings. Results show that the GA based search is significantly superior to standard methods.

6.2 Interactive GA Jazz

"*GenJam*" [3] is a nice twist on the preceding examples, using the Ears rather than the Eyes of the human operator to choose the preferred individuals from the generated population.

John A. Biles describes GenJam as "*a genetic algorithm-based model of a novice jazz musician learning to improvise. GenJam maintains hierarchically related populations of melodic ideas that are mapped to specific notes through scales suggested by the chord progression being played. As GenJam plays its solos over the accompaniment of a standard rhythm section, a human mentor gives real-time feedback, which is used to derive the fitness values for the individual measures and phrases. GenJam then applies various genetic operators to the populations to breed improved generations of ideas*".

6.3 Interactive Computer Art

There are several examples of interactive computer 'Art' programs where pictures are generated by a GA, and the human viewer can then choose the most interesting members of each generation for onwards evolution. One Internet site [14] provides a quick overview of the concepts and lists some of the practitioners in particular Todd and Latham's *Evolutionary Art and Computers* [16].

The Carnegie Mellon site [15] allows on-line Internet interaction with the Art Generator program using a 'voting' system of input by numbers of users to select the 'fittest', or in this case, 'most appealing' pictures for further generation. The site includes an overview of the process and provides references, including Karl Sims' work [17].

6.4 Interactive Evolution

The paper by Wolfgang Banzhaf [6], discusses the concept of "Interactive Evolution", which involves the use of a human operator on-line to interact with the selection phase of the GA. The problem that is being addressed is that of selection conditions that are not formalizable, and need to make use of subjective human judgements. The user is presented with a graphical display of the potential solutions and makes his selection which defines the members of the population to be taken forward to the next generation of the GA.

Banzhaf also introduces the idea of using a predefined fitness criterion to present the potential solutions in a pre-sorted order to assist the human user in sifting through the data. Other issues discussed are the problems with humans selecting from a large number of options - he suggests that ten choices is a suitable maximum.

Further developments that are proposed include the use of other human senses (smell, touch etc.) as well as the obvious ones of sight (see [2]) and hearing (see [3]).

7. The Hybrid Fitness Function

We will now consider the potential for using a human GUI to interact with the GA fitness function, such that the system delivers a ''Hybrid" function consisting of both computed values and the subjective human input.

The GA will create each generation using the bit strings or other representations, and each individual will have some properties that can be measured. In the case of the Multi-Project Resource Scheduling GA, the individuals have properties such as Change in Start Date, and Change in overall Duration. The 'FacePrints' package uses parameters that define the position of the facial image in multi-dimensional face space.

7.1 Initial Sorting

Given a population of 10s or 100s of individuals, it becomes a time-consuming operation to sift the good from the bad. Indeed, Wolfgang Banzhof [6] believes that over about 10 choices the human operator cannot make coherent selections.

It is therefore necessary to have some order applied to the individuals prior to the human selection. We can use any generated or derived parameters with a predefined sort order, or let the user decide the weighting on the parameters to adjust the sort order to his preference.

For example, we could select the individuals in our Resource Schedule by order on the Change in Start Date alone, or we could take some proportion of each of the parameters. The first element of the Interactive Fitness Function is therefore a mechanism for the user to define (or play with) the weighting factors which are to be used to develop the pre-sorted sequence of the individuals for presentation to the user.

7.2 Representation

The next issue with any interactive GA is the ability of the system to generate a satisfactory representation of the generated individuals so that the human operator can make his choices.

This assumes a straightforward mapping of the individual to the 2D (or pseudo 3D) images capable of being displayed on the computer screen, or potentially the sound or music through the PC speakers.

For GenJam, the PC speakers allow access to the user's ears, and in FacePrints the system creates the images of the generated faces. Dawkins 'Biomorphs', Spider's Webs and arthromorphs are all designed to be 2D screen images.

For graphical images, many can be displayed at once (allowing for Banzhof's reservations) with perhaps squares of 9, 16 or 25 images. The resolution of the PC screen can also be a limiting factor in displaying images of high enough quality to be distinguishable one from another. Aural GA output (sound-bites ?) is by nature a serial process, one after the other, with the user needing to move back and forth between the individuals.

In the case of our Resource Scheduling problem, and as already used in the Interactive AI based schedulers ([7] and [8]), the most appropriate displays are the Gantt (or Bar) chart with horizontal bars representing the dates and duration of tasks against a timescale, or the Resource Profile showing Resource usage accumulated by date.
These can take the form of a high level 'overview' with just an outline of the shape, or detailed task-by-task representation. Experimentation will be required to determine the level of resolution needed for the user to be able to make satisfactory choices, with 'zoom' capabilities for detailed examination of selected individuals.

The Interactive Evolution technique is therefore limited to applications that lend themselves to Graphic or Aural displays, or to the ingenuity of the designer to represent the individuals successfully.

7.3 Selection

The human user must be able to easily interact with the GA and select his choices. We assume a Windows-style interface with Drag-And-Drop, Point-And-Click Mouse control. In the case of a pictorial display the user can directly select the required individuals by Clicking on the image. In the case of Aural input, the user must have some graphical representation of what he is listening to so that he can select "Like" or "Dislike" for each sound-bite.

He must also be able to move between the images, reselect ones he has discarded after a change of mind, and perhaps be able to reset the sort weighting to re-order the images during the selection process.

8. Implementation

Various existing systems (some GA, some AI, some Artistic) have demonstrated successful interaction between the computer program and the human operator's subjective selections. They take advantage of modern PC systems with Windows-style interfaces which provide a suitable environment within which the human and PC can interact.

The success of this approach depends on the ability of the system to represent the generated populations in a distinguishable manner (usually graphical) for the human user to determine his selections.

8.1 The Interactive Algorithm

We now have the basis for developing an Interactive GA. The elements involved are:

- Standard GA population generation.
- Well defined and suitable graphical (or aural) representation of the individuals.
- One or more internal parameters defining the fitness for the pre-sort process.
- Optional user defined parameter weighting to pre-sort the displayed images.
- Pre-sorting of individuals on the display.
- Interactive selection and re-sorting by the user.
- Standard GA recombination, mutation and regeneration procedures.

An outline of the Interactive GA process is shown in the box below, :

```
Initialise the Population
Input User Pre-Sort Weighting
While Not Finished
        Determine Internal Fitness Parameters
        Display the current Population in Pre-Sorted Order
        Input User Fitness Value Changes
        Evaluate Fitness Order
        Breed next Generation
Repeat
```

8.1.1 Determine Internal Fitness Parameters

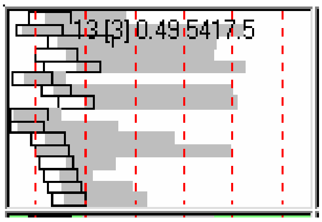
The algorithm determines the basic fitness based on the two parameters:

- The slippage of the start date of the project in days.
- The extension of the project's overall duration in days.

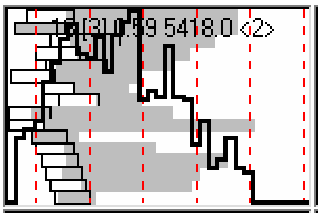
These two values are summed to provide the basic system fitness value. The fittest solution is the one in which there is the least slippage in these values, and therefore the smaller the number, the higher the fitness value.

8.1.2 Display the current Population in Pre-Sorted Order

During development, the program initially displayed a graphical representation of the solution in terms of a GANTT chart, where the start and end dates of the projects define the span of a horizontal bar against a timescale: (This is one element of the display of the set of solutions).



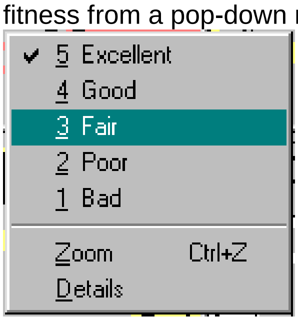
Trials with project manager's showed that although this was useful and that they could make fitness choices from the GANTT chart, a better display included a Resource Profile, derived from the resultant solutions:



The resource profile shows the cumulative resource usage over time, mapped on top of the GANTT display. The Project Managers found this a much easier way of determining the relative fitness of the displayed solutions.

8.1.3 Input User Fitness Value Changes

The user reviewed the set of solutions (see figure 9.1 below) and then changed the fitness values as necessary by Right-clicking with the computer mouse and selecting the required fitness from a pop-down menu:

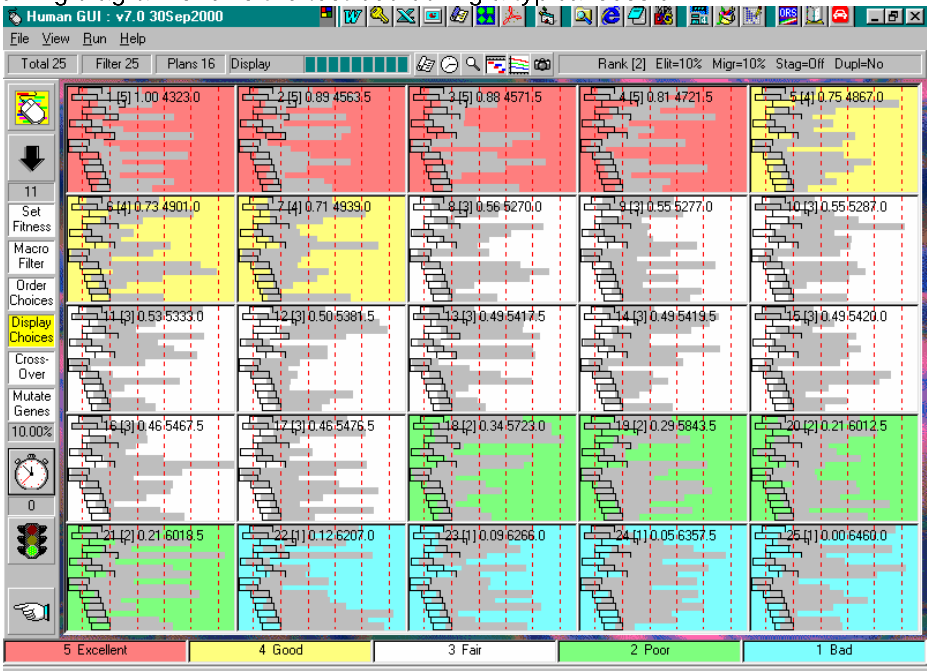


9. Results

As part of the ongoing research on which this paper is based, a computer program has been written as a test-bed to experiment with the various techniques of providing a human GUI to a scheduling GA.

Initial experiments worked with providing a GANTT (bar) chart based GUI, where the initial and scheduled positions of the project summary bars were displayed for the user's fitness selection.

The following diagram shows the test bed during a typical session:



The 25 boxes each represent one of the GA solutions derived from the evolutionary process. The colours are used to group the GA calculated fitness prior to the user's choice, the system uses 5 groups of fitness derived from the overall fitness factor.

This display is then used by the Project Manager to choose and specify his preferred fitness values.

Results

1. Basic Algorithm compared with standard serial scheduler (MS Project) - 16 project plans

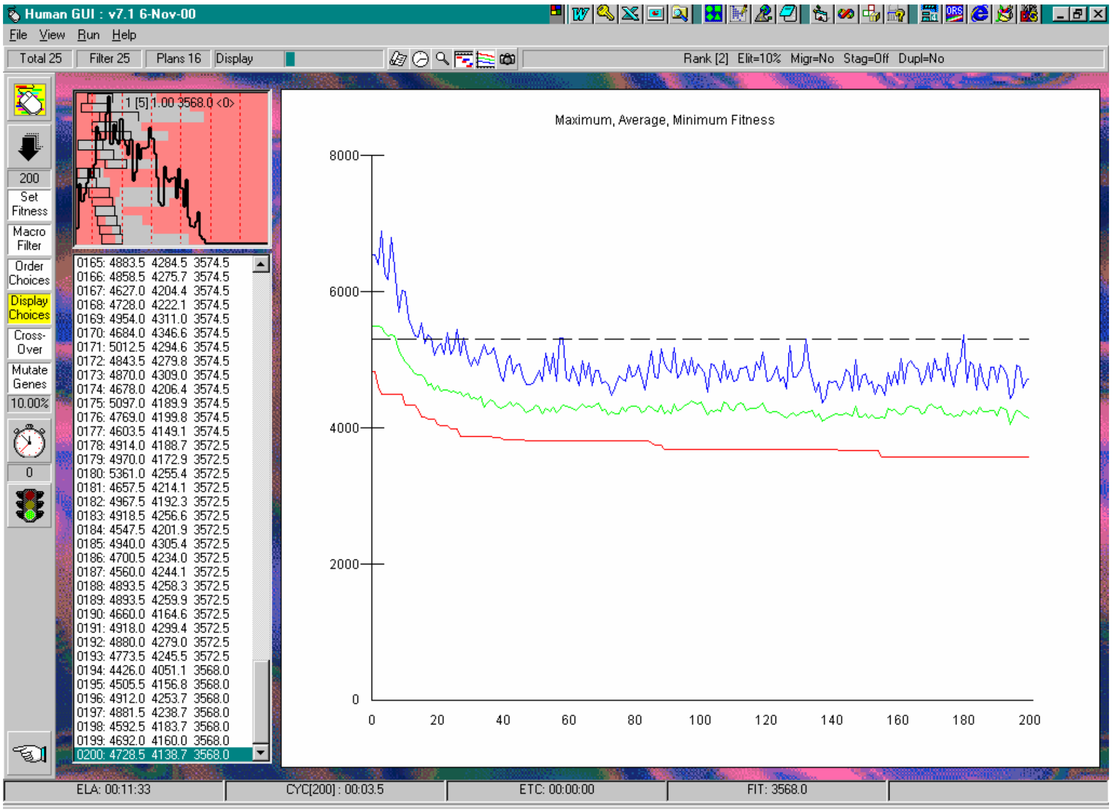
GA program		MS Project
3874		5305
3980		
3345		
4023		
4079		
3800		
3897		
3927		
3910		
3912		
3640		
3940		
3568	Example 1	
4054	Example 2	
3753	Example 3	

Values are total days slipped from original start

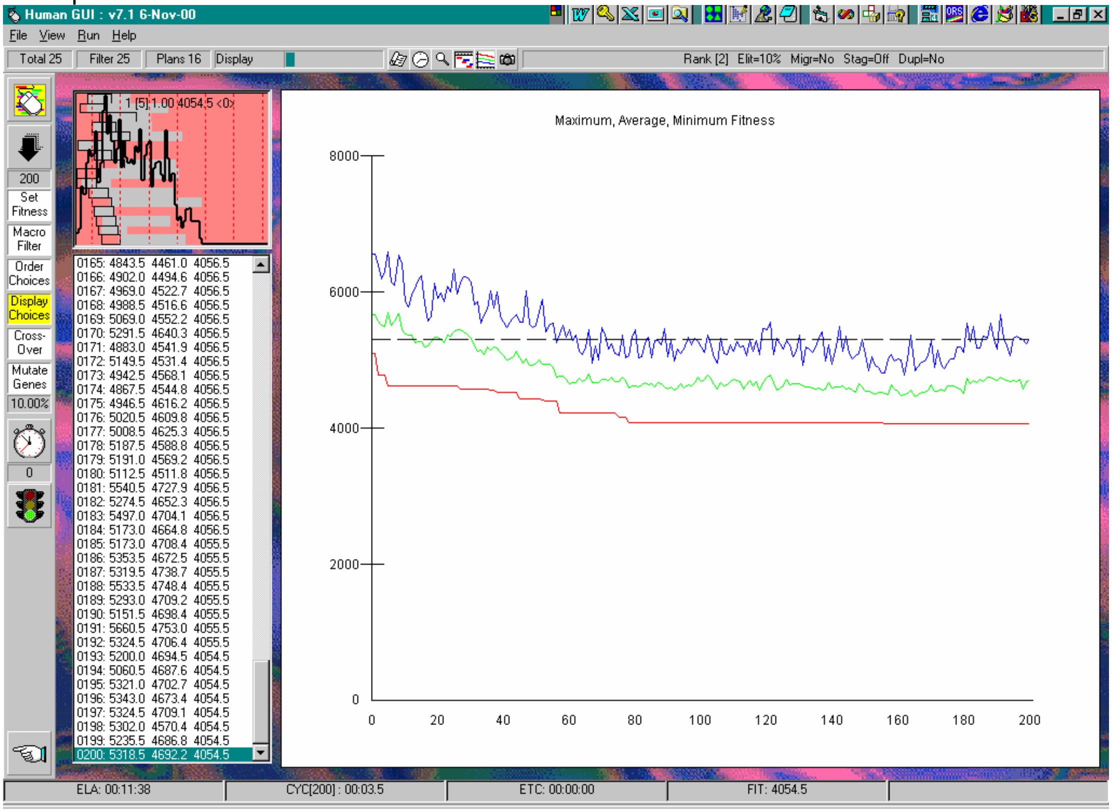
= SUM (Start Date Slip + Overall Duration Slip) for each project plan

The following diagrams show the final results for the last three tests in the table above. The black dashed line shows the MS Project serial schedule result, whilst the lower Red line shows the Fittest (lowest slip) GA result.

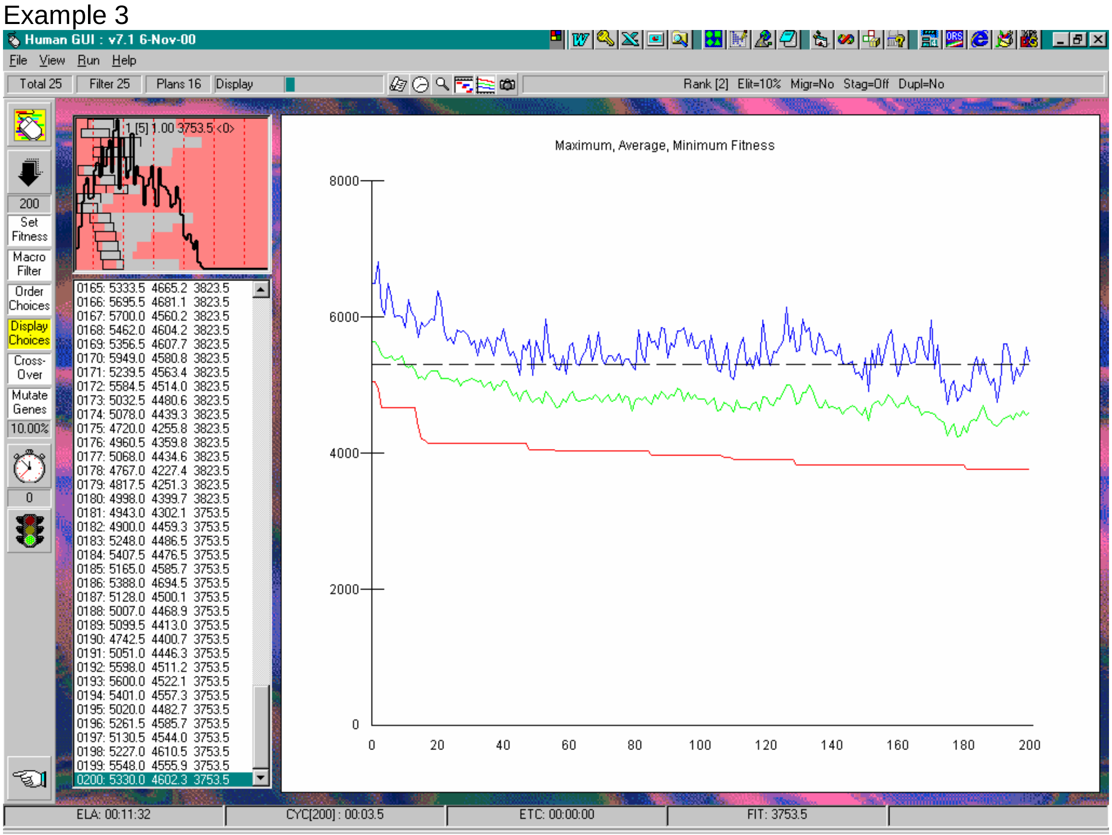
Example 1



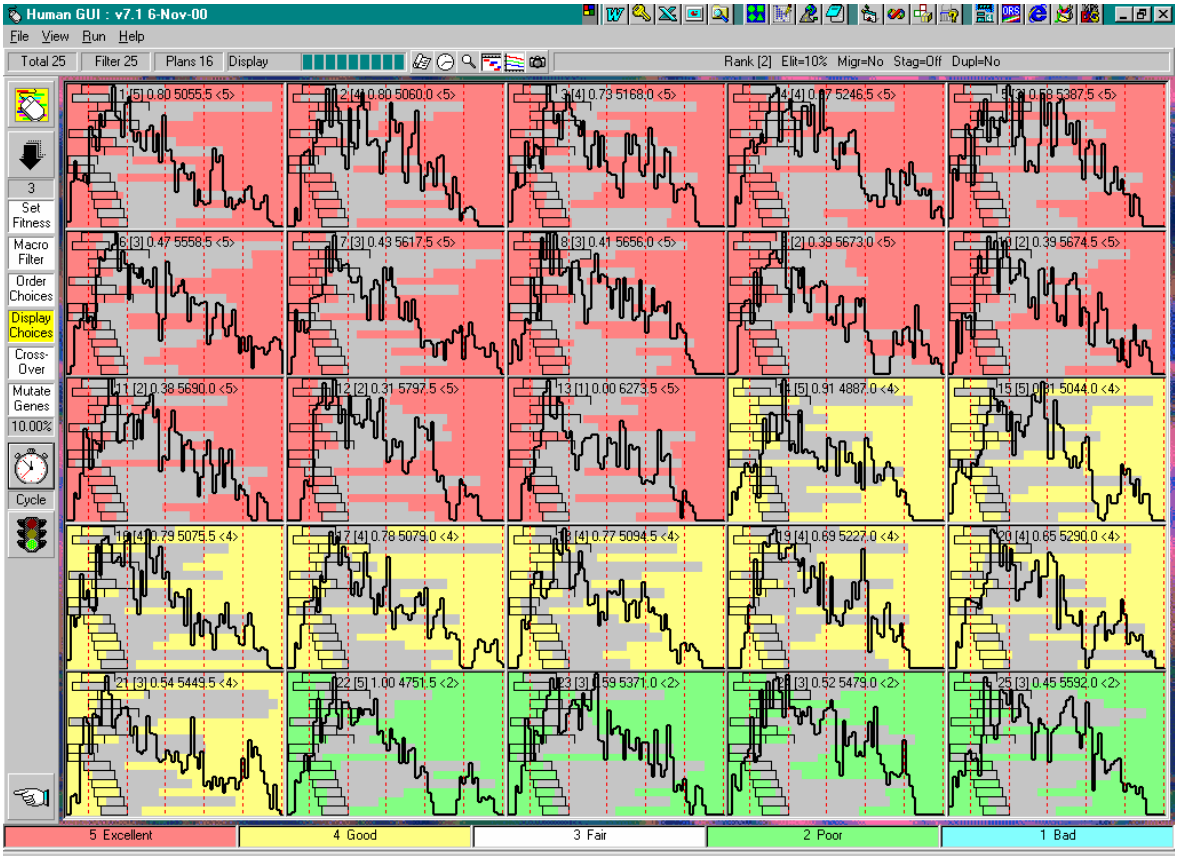
Example 2



Example 3



The following screen shot shows the result of using the Project manager's input to drive the result to a Diagonal Line. This is for the case where a Project Manager knows that new projects will be coming in over the next period, and wants to gradually phase out the usage of resources, so that resources become available for the new (but as yet unknown/unplanned) projects. The top left diagram is the "Fittest" solution as selected by the User's criteria, and has a very different shape to the best solutions found by the basic algorithm in the diagrams above.



10. Summary

This paper has described the design and implementation of a GA based system to Schedule Resources in a Multi-Project environment.
The main aim was to include an element of Human interaction within the GA Fitness function, so that a Project Manager's 'gut feel' and implicit knowledge could be used to direct the solution process without requiring any complex definition language or expert system data gathering operation.
The experiments have shown that the system can be directed by the human operator, so that the solutions 'bred' by the GA tend towards a shape or profile preferred by the user.

10.2 Applications

Systems that could use this technique (i.e. involve some graphical aspect) include:

- Industrial and Packaging Design (vehicles, furniture, white goods)
- Scheduling systems (Factory, Job shop, Project Management)
- Pattern design (Wallpaper, Flooring, Carpets)
- Police Investigations (Photofit Faces, determining the make of 'Get-away' Cars etc)
- Environmental Design (landscaping, shopping malls, room layouts)
- Advertising (Logos, Slogans, Brochure Layouts)

10.3 Issues

Particular issues that need to be considered when developing such an Interactive GA include:

- Can the individuals be represented graphically (or for some other sense) ?
- Are the individuals distinguishable by the user. Is the resolution high enough ?
- How many individuals can a user successfully choose from - 10 or many more ?
- Are some internal parameters available for the pre-sort ?

10.4 Conclusion

Taking into account the potential issues, it is entirely feasible to use existing PC technology to deliver Interactive Graphical Interfaces to standard GA processes. In particular, the Multi-Project Resource Scheduler would benefit highly from interaction with a human user to enable the important but subjective fitness parameters to be specified.

The ability of an Interactive GA to allow subjective human interaction with the fitness functions will also allow the GA as a methodology to become more of a user-friendly tool, rather than a 'black-box' application using arcane processes to deliver results that may be difficult to interpret or believe. Computer systems in general are more acceptable when seen as an extension of the human rather than as a replacement for them.

References

[1] *Genetic Algorithms in Search, Optimisation and Machine Learning*, David E. Goldberg, Addison-Wesley, 1989.

[2] *Tracking a criminal suspect through face space with a genetic algorithm*, Victor S Johnston and Craig Caldwell, G8.3, Handbook of Evolutionary Computing, 1997.

[3] *GenJam: A Genetic Algorithm for Generating Jazz Solos*, John A. Biles, Rochester Institute of Technology.

[4] *The Blind Watchmaker*, Richard Dawkins, 1986

[5] *Climbing Mount Improbable*, Richard Dawkins, 1996

[6] *Interactive Evolution*, Wolfgang Banzhaf, C2.9, Handbook of Evolutionary Computing, 1997.

[7] *Constraint-Oriented Cooperative Scheduling for Aircraft Manufacturing*, Patrick Esquirol and Pierre Lopez, IEEE Expert, Jan-Feb 1997

[8] *Knowledge-Based Scheduling systems in Industry and Medicine*, Jurgen Sauer and Ralf Bruns, IEEE Expert, Jan-Feb 1997

[9] *A Genetic Algorithm White Paper*, Salvatore R. Mangano, [INTERNET]

[10] *Genetic Algorithms: An Overview*, Melanie Mitchell, COMPLEXITY, Wiley, 1995

[11] *A Genetic Algorithm Tutorial*, Darrell Whitley, Colorado State University, 1993

[12] *Genetic Algorithms: Principles of Natural Selection Applied to Computation*, Stephanie Forrest, SCIENCE, Vol.261, Aug 1993.

[13] *The Visual Display of Quantitative Information*, Edward R. Tufte, Graphics Press, 1983.

[14] *What Is Evolutionary Art ?* Andrew Rowbottom, [INTERNET]

[15] *Instructions for International Interactive Art*, Carnegie Mellon School of Computer Science, [INTERNET]

[16] *Evolutionary Art and Computers*, Todd and Latham, Academic Press

[17] *Artificial Evolution for Computer Graphics*, Karl Sims, Proceedings of SIGGRAPH 1991.