VAST improvements to Diagrammatic Scheduling using Representational Epistemic Interface Design

David Ranson and Peter C-H Cheng

Department of Informatics, University of Sussex, Falmer, BN1 9RN, UK
d.j.ranson@sussex.ac.uk, P.C.H.Cheng@sussex.ac.uk

Abstract. REpresentational EPistemic Interface Design (REEP-ID) advocates exploiting the abstract structure of a target domain as the foundation for building cohesive diagrammatic representations. Previous research explored the application of this approach to the display and optimisation of solutions to complex, data rich, real world problems with promising results. This paper demonstrates the application of these principles to generate interactive visualisations for solving complex combinatorial optimisation problems, in this case the University Exam Timetabling Problem (ETP), entirely. Using the ETP as an example the principles of REEP-ID are applied, illustrating the design process and advantages of this methodology. This led to the implementation of the VAST (Visual Analysis and Scheduling for Timetables) application, enabling individuals to solve complete instances of the ETP using interactive visualisations. Rather than using automated heuristics or algorithms, VAST relies entirely on the user's problem solving abilities, applying their knowledge and perceptiveness to the interactive visualisations maintained by the computer. Results from an evaluation of VAST support the use of the REEP-ID methodology and the case for further research. In the closing discussion these findings are summarised together with implications for future designers.

Keywords: Representational Epistemic Design, Exam Timetabling, Visualisation, Diagrammatic Reasoning

1 Introduction

Educational Timetabling is a vibrant research field and also a very practical problem faced yearly by almost all educational institutions. Most research on this topic rightly concentrates on the incremental improvement of automated systems and the heuristics used to solve timetabling problems. Typically institutions rely on automated approaches to generating exam timetables; however the solutions generated by such automated systems do not always match real world requirements of a timetable and existing interfaces can make it difficult for practitioners to make minor intuitive changes to a solution without the risk of damaging the timetable in unforeseen ways.

Here, an alternative approach is presented that applies a REpresentational EPistemic approach to Interface Design (REEP-ID). The REEP approach advocates analysing the underlying conceptual structure of a problem domain and finding graphical schemas to directly encode that structure [DPhil Thesis1]. The approach has been used to design novel representations for science and mathematics learning (e.g., [2]) and to create novel interactive visualisations for event scheduling, personnel rostering and production planning and scheduling (e.g., [1, 3, 4]). This paper is a contribution to the wide research programme on the REEP approach and it is in particular exploring how REEP-ID can be used to create novel representations that enable solutions to instances of the ETP to be found from scratch by humans without resorting to major computer automation of the problem. It builds upon the STARK-Exam [3] systems that successfully support the problem of manually improving existing examination solutions. By combining the problem solving and perceptual abilities afforded to humans with well-designed interactive visualisations maintained by a computer, the aim is to demonstrate certain advantages over purely automated approaches and existing scheduling interfaces. It will be shown that the application of a REEP-ID approach provides one framework that can be applied to generate interactive visualisations capable of supporting these goals.

The following section introduces the ETP in more detail. After this the development of the Visual Analysis and Scheduling for Timetabling (VAST) system is presented in Section 3, together with a discussion of how the REEP-ID methodology was applied. Section 4 summarises some of the results from an evaluation of the VAST system and the paper closes with a brief discussion of the implications of this approach.

2 Exam Timetabling

Many timetabling problems have similar conceptual structures, instances of educational timetabling problems such as the ETP usually differ between institutions making theoretical models and algorithms harder to generalise. As a special case of educational timetabling the ETP is interesting to focus on for several reasons, it is a practical problem that has been the focus of much research for at least four decades. Improvements made to exam timetabling system interfaces could have benefits to practitioners, researchers and students sitting the exams. When scaled up to real life situations the size and algorithmic complexity of the problems make finding valid solutions very difficult. The conceptual complexities inherent in the problem, such as those arising from the sheer number of entities in a typical ETP and the relationships between them, make it difficult for individuals to conceptualise complete instances of the ETP with current widely used interfaces.

2.1 Definition

Fundamentally timetabling is the assignment of events to time periods subject to constraints. In many cases the events also need to be assigned physical locations where they will take place. In the timetable model adopted here [5, 6] a complete ETP representation consists of some abstract timetabling *Problem*, e.g. the University Exam Timetabling Problem, a concrete *Instance* of a problem which defines the available resources, events, time periods, unique constraints and finally a *Solution* or Solutions to an Instance that stores the actual assignment of events. In the case of the ETP there are many concepts, with associated attributes, that need to be represented: Time Periods, Rooms, Students, Exams, the set of Constraints that determine what assignments are valid, and the assignment of solution variables to exam events are all examples of these.

There are also logical relationships between entities in the ETP, for example sets of exams that have the same duration or that share common students, or *Clusters*. Grouping exams into such Clusters has the benefit that all Exams within a Cluster can be scheduled into a single time period, assuming there is sufficient room capacity. Exams that do share common students are termed *conflicting* exams.

2.2 Constraints

Within the ETP constraints can be either *hard* or *soft*. To be feasible a solution cannot violate any hard constraints, whilst it is undesirable to violate soft constraints they can exist in a feasible solution. It is the goal of exam timetabling to eliminate all hard constraint violations and as many soft violations as possible. Table 1 summarises the most important common constraints that are typically found in the ETP [7]. The *Conflict* constraint, together with *Room Capacity* and *Time Period Duration* constraints are present in all instances of the ETP, we also include *Completeness* as a hard constraint indicating that *all* exams need to be assigned for a timetable to be feasible. The precise details of the other constraints will differ between institutions and this is not an exhaustive list.

Table 1. Common constraints found in most instances of the ETP.

Constraint	Description	Hard/Soft?
Completeness	All exams must be timetabled	Hard
Room Capacity	An exam must be timetabled in a location with sufficient capacity for all the assigned students.	Hard
Time Period Duration	An exam must be timetabled in a time period of sufficient duration	Hard
Conflict	A student cannot sit more than one exam simultaneously.	Hard
Consecutive	A student should not sit more than two or more exams in immediate succession. Depending on the institution this may or may not include overnight gaps.	Typically soft
Order Precedence	An exam should be scheduled before, after or at the same time as another exam.	Depends on instance
Mixed Durations	Exams in the same room should be of the same duration to minimise disruption	Typically soft

To allow an automated solver to evaluate the quality of a solution each constraint violation is typically applied some weight which can then be aggregated to calculate an associated penalty cost for that solution. In this situation an optimal solution with no violations would have a cost of zero whereas a solution with many hard constraint violations would have a very high penalty cost. When using the VAST visualisations this cost function is less important to the user than the more meaningful visual representation of the constraint violations encoded in the interface possibly combined with other knowledge of an ideal solution not taken into account by the formal evaluation function, e.g. political influences.

2.3 Existing Interfaces

Existing interfaces, including the most widely used commercial applications[8], for inspecting and modifying Exam Timetables are typically based around a traditional timetable layout with days spread across the horizontal dimension and time periods within each day across the vertical dimension. This approach is undesirable from a representational point of view as the concept of time is mapped to two different spatial dimensions of the timetable layout.

This traditional layout, although good for quickly searching for an exam using a day and time as an index, makes some judgments useful to solving these problems very difficult to make, for instance: How much free room capacity remains at any time?, What are the effects on the global timetable of moving a single exam? It is hoped that by taking a representational epistemic approach many of these important concepts, together with the features of the problem that support effectively solving instances of it, can be encoded into a more cohesive representation.

There are also other drawbacks to many of the existing interfaces currently used, information important to conceptualising the problem, such as exam location, violation of constraints or exam durations, are often only represented through text labels or lists. To gain a complete global view information often has to be accessed through unique dialog boxes or by switching between different perspectives, with multiple representations within the interfaces.

3 Representational Epistemic Interface Design

The REEP-ID approach to interface design provides four main steps in the design of a novel interface:

1. Identify entities in the target problem domain and the basic relationships between them.
2. Find the conceptual dimensions, sources of conceptual complexity and levels of granularity present in the problem.
3. Identify the primary conceptual dimensions and try to find a cohesive visual/spatial mapping of these.
4. Design concrete visual representations for the remaining entities and relationships that respect the mapping of the primary conceptual dimensions.

The start point for the design of VAST was our previous STARK-Exam systems [3]. The overall goal was to design a representation that effectively captures the conceptual features and relationships of the ETP without altering or adding to the existing complexities. The first steps in the new design involved making explicit the different levels of granularity of entities and concepts that were implicit in design of STARK-Exam systems. These features of the ETP, based on the entities and relationships identified in the existing ETP model [6] introduced in Section 2.1, contribute to the conceptual complexity of the domain and are summarised in Table 2. At the operational level users will interact with the representation, introducing individual differences and understandings that affect the operational complexity. The functional level contains the features that make the problem so difficult to solve, for example the dynamic interactions between constraints and entities. The molecular level shows some conceptual structures that can emerge from the relationships and groupings of features found at the atomic level, these structures are inherent to the abstract problem so should also be available in an effective closely-mapped representation. Finally the atomic level shows the entities that the representation is attempting to model to solving the ETP, complexity at this level can be found in the magnitudes of the dimensions and sheer number of entities and functional relationships that can exist in a typical ETP.

Table 2. Different levels of granularity of Conceptual Complexity identified in the ETP

Level	Concepts
Atomic	Students, Exams, Rooms, Times
Molecular	Clusters, Departments
Functional	Conflicts, Capacity, Availability Order, Duration, Completeness
Operational	Heuristics, User Interaction

Once the features of the target domain had been identified, together with the level of granularity at which they contribute to the conceptual complexity of the problem, a cohesive mapping of the primary conceptual dimensions had to be found to the provide the foundations to a *global interpretive framework[1]*. The primary conceptual dimensions for the ETP solution were identified as *time periods* and *room capacity* and the mapping is the same used within the STARK-Exam system, and is presented in Figure 1. Other mappings such as the traditional timetable layout were briefly considered but rejected for the reasons previously discussed. If suitable representations for crucial features or concepts in the problem could not be found that fit the visual spatial mapping of the primary conceptual dimensions, then this would be an indication that these mappings might need to be rethought. The following sections describe the particular graphical schemes that have been used to encode each of the levels and concepts described in Table 2.

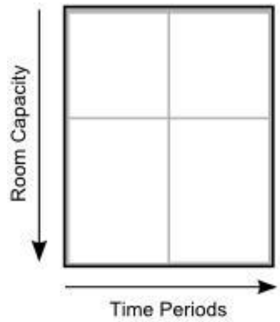


Fig. 1. Mapping the Primary Conceptual dimensions *Time Periods* and *Room Capacity* to a visual spatial representation. Room capacity is represented in the vertical dimension and time periods are shown across the horizontal dimension in chronological order. Breaks in the background indicate the start of a new time period or the start of a new room.

Within the interactive visualisations being developed every graphical entity representing some concept from the problem domain is termed a *glyph*. Each type of glyph should be clearly distinguishable from others and have a clear topological and geometric form. To help maintain a cohesive unambiguous design throughout the visualisations the geometric structure of a glyph cannot change and each glyph always represents the same information, for example once an exam has been encoded as a particular glyph it cannot then be represented by any other glyph or representation within the same visualisation. Interaction with the visualisation is determined by the interactive capabilities of each glyph.

Whilst this visual structure of a glyph cannot change, a glyph can be *decorated* to augment it with further information. Decorations can be used to show different perspectives of information, or perhaps making extra information available in response to some user interaction. Colour is typically used as a decoration on a glyph and as such the colours can change to indicate concepts such as group membership (cluster, department etc), to highlighting a glyph to show it has been selected by the user or if a user's attention needs to be brought to a particular glyph.

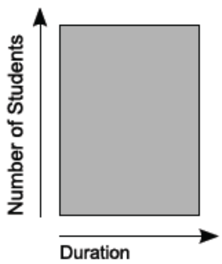
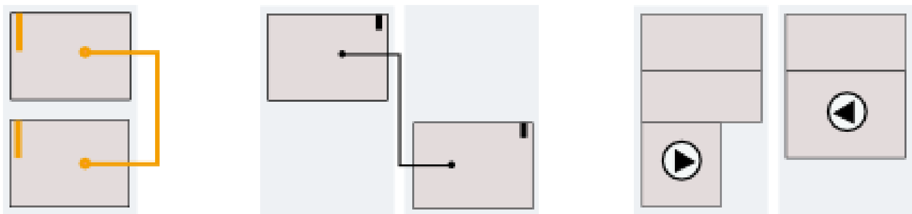


Fig. 2. An Exam Glyph, a concrete graphical representation of an Exam exploiting the spatial mapping of the primary conceptual dimensions. The width is always proportional to the duration of the Exam whilst its height is proportional to the number of students sitting that Exam.

The first glyph we explicitly introduce is the Exam Glyph, shown in Figure 2, this is a visual representation of the Exam event that exploits the mapping of the primary conceptual dimensions to encode the number of students sitting the exam (height along the vertical room period dimension) and the duration of the exam (as width along the horizontal time period dimension). The time period and room location of an assigned exam will be encoded as the glyphs position in the solution view explained in Section 4.2.

In Table 1, and again when examining the sources of functional conceptual complexity, we identified some of the common ETP constraints that need to be encoded in the representation. In this section we show how some of these constraints are already represented by the following the syntactical rules provided by the spatial, geometric and topological mapping of the primary dimensions and exam glyphs, whilst others require new glyphs to be added to the visualisation. For each of the constraints a number of representations were considered such as different forms of lines between violating exams, Shading violating exams, or icons decorating violating exams.



(a) (b) (c)

Fig. 3. Representation of constraint violations in VAST. Sub-figure (a) shows a conflict violation between two exams with a thick orange line and size bars on the left to indicate the proportion of students in the conflict. Sub-figure (b) illustrates a consecutive violation between two exams in successive time periods with a thin black line and size bars on the right. Finally Sub-figure (c) shows both a mixed duration violation, highlighted by the jagged right hand edge in the time period on the left, and an order precedence violation between two exams indicated by arrowheads pointing in the direction the exams should be moved to resolve this violation. These violation representations can always be perceived even when applied, in any combination, to the same exams or otherwise overlap.

The room capacity and time duration constraints are encoded by the consistent mapping of time and room in the primary dimensions and the exam glyphs. An exam glyph will not physically fit in a room time period combination in the timetable layout shown in Figure 1 unless that location has sufficient duration or room capacity for that exam. Also the mixed duration constraint is represented by the alignment of the right hand edges of exam glyphs, if exam glyphs in a timetable location have a jagged right hand edge, as illustrated in Figure 3(c), a violation of the mixed duration constraint is present.

It was found that lines were useful to link violations that were spatially close, for example conflict and consecutives violations, shown in Figures 3 (a) and (b), only occur in the same and successive time periods respectively. Drawing lines between points that are not always spatially close, for example for order precedence constraints, adds to visual clutter and it is not always easy to quickly determine which the violating exams are. Care also needed to be taken to ensure that lines do not obscure other featured of the visualisation, in VAST all lines follow defined paths between exams and different violations have different thickness so they can always be perceived.

Decorations are be applied to conflicting exam glyphs designed to show the magnitude of conflict and consecutive violations, information necessary to strategically resolve such violations. In this scenario a *size bar* is shown which indicates the proportion of students in an exam that contribute to the violation, this allows judgements to be made as to which are the most conflicting exams in a violation. Also order precedence violations are represented by an arrowhead icon, the initial intention had been to represent these with arrow connectors however, as discussed, we wanted to avoid the use of lines except in spatial localities. To resolve this issue we used the arrowhead decoration on violating Exam glyphs to indicate the temporal direction the exams should be moved in order to resolve the conflict, this is illustrated in Figure 3(c).

Shading of exams to encode constraint violations was also avoided as coloured shading was defined as a decoration that could be changed to indicate group membership and selection to the user. Shading, in particular an easily distinguished hatching pattern, is however used to highlight all exams that *conflict* with the user's current exam selection. This helps resolve constraints intelligently as users have immediate feedback highlighting where an exam can be moved without introducing further violations. An extension of this, dynamic time period restriction, also shades out entire time periods where scheduling an Exam would cause further violations to be introduced and is illustrated later in Figure 6.

4 VAST

4.1 Instance Visualisation

The next two sections show how the glyphs and that have been described are combined in these two interactive visualisations to completely represent the ETP. The VAST system is made up of two main components, corresponding to the *Instance* and *Solution* parts of a timetable model introduced in Section 2.1. A user can switch between the Instance and Solution components at any time by changing between standard tabbed pages and a dialog could be opened to show the glyphs that were currently selected in the instance view if necessary, it might be beneficial to instead show both visualisations on two monitors simultaneously. The purpose of the instance visualisation is to represent exams before they are assigned to the timetable, in this early stage of the problem solving process the concept of time periods is of little help in structuring the ETP entities as no assignments exist.

Rather than presenting this information in an arbitrary list the exams are instead grouped, by columns, into *clusters*. Exams can be selected for scheduling either by clicking individual exams, lassoing any group of exam or by selecting entire clusters by clicking a handle at the top of each column. The intention of cluster selection is to provide a simplification of the conceptual problem to the user. These entire columns can be assigned to single time periods in the solution without introducing any conflict constraints. Given a good enough clustering of the exams and enough space in the timetable a feasible solution could be constructed by assigning each of these columns to different time periods, eliminating the otherwise tedious necessity to consider every individual exam.

The instance view also supports other groupings of the exams by attributes such as department membership or duration. Columns can also be sorted by the number of exams they conflict with, the number of students in the exams or the duration of exams. These groupings and orderings, based on conceptual features of the abstract problem, allow users to make intelligent choices about which exams to select to implement different problem solving strategies.

The final use of the instance view is to represent the completeness constraint, identified in Table 1, that every exam must be assigned to the timetable. As can be seen in Figures 4 and 5 once an exam has been scheduled it is moved from the Instance visualisation to the Solution visualisation however its outline remains in the instance view. This allows users to quickly perceive how many, and precisely which, exams are waiting to be timetabled.

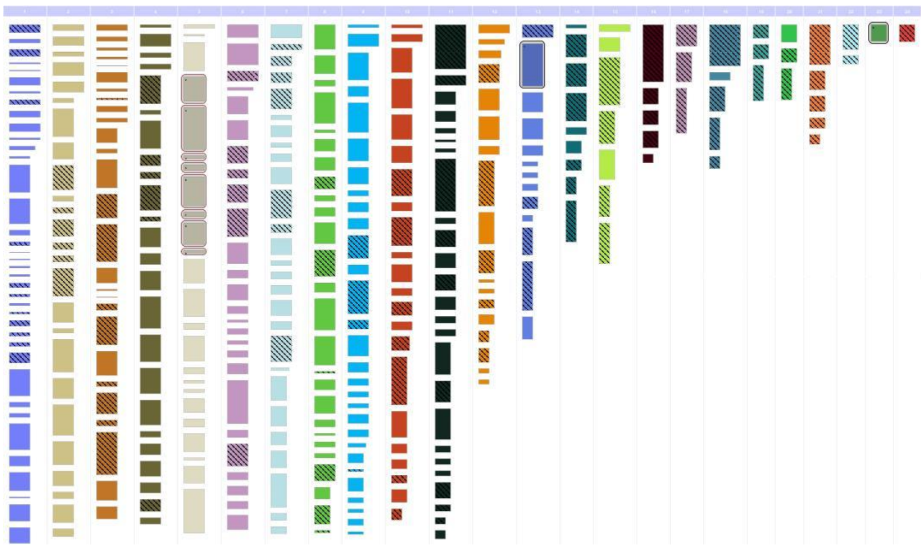


Fig. 4. The VAST Instance visualisation showing unscheduled exams grouped into columns such that exams in the same column do not conflict. The columns can be changed to group or the exams by other attributes such as department membership or their duration. Exams that been selected by the user are surrounded by a coloured opaque lozenge and exams that share common students (conflict) with the current selection are hatched.

Fig. 5. The Instance visualisation once all exams have been assigned to timetable locations. Within each outlined exam is a reference to the Room and Time Period that the exam has been assigned to so it can easily be located within the Solution visualisation. An exam reappears here if it is unassigned from the timetable solution.

4.2 Solution Visualisation

The second and most important component of the VAST interface is the Solution Visualisation shown in Figure 6. Time periods are shown across the horizontal dimension grouped into days with gaps representing overnight gaps; rooms and their capacities are represented by the vertical dimension. Most of a users time will be spent in this visualisation, it supports the initial assignment, perception of violations and subsequent optimisation of ETP solutions.

Within the solution view selected Exams and groups of Exams can be assigned by clicking in the appropriate time period; for this initial assignment exams are automatically located in the smallest room they fit in so that groups can be quickly assigned without worrying about individual room allocation. Exams are then moved by clicking first an exam and then its destination room/time period location in the timetable. After selecting an exam to move all conflicting exams are hatched, and time periods containing conflicting exams are shaded, this has the effect of highlighting the time periods that selected Exams can be assigned to without creating new violations. Exams can also be unscheduled and entire time periods of exams can be swapped by clicking on handles at the top of each column.

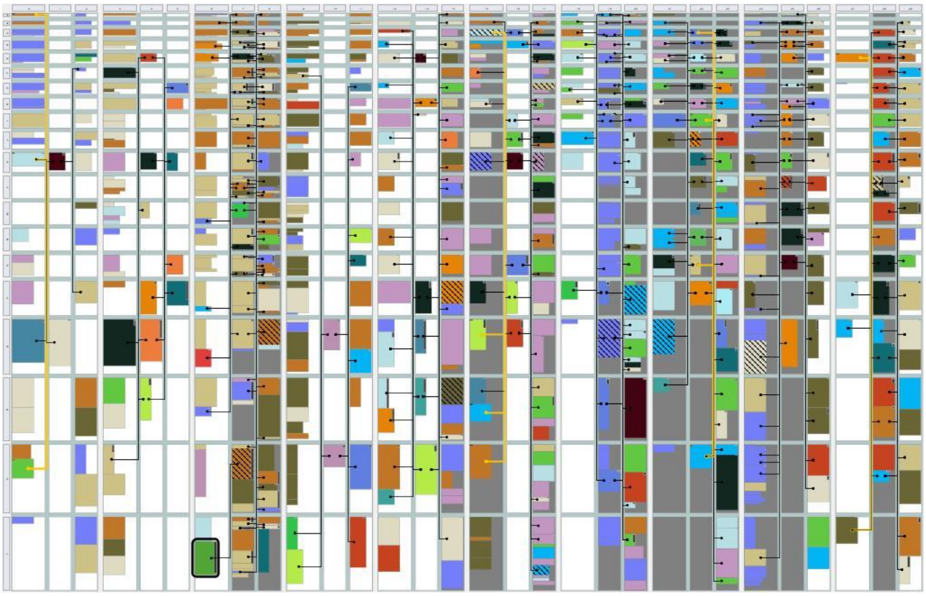


Fig. 6. Solution Visualisation showing a complete solution to an ETP instance. Both conflict and consecutive violations are present in this solution as indicated by the orange and black lines respectively. As before the currently selected Exam, in the lower left quartile of the image, is shown surrounded by an opaque lozenge and conflicting exams are hatched out. The grey time periods are the "restricted" time periods and moving the current selection to these locations, or next to these locations, would introduce new conflict or consecutive violations respectively.

5 Evaluation

An evaluation of the VAST system was undertaken to determine how successful the interface was, in particular how effective the representations were in assisting users to conceptualise and solve ETP instances. An on-line competition was created allowing participants to compete for a £70 prize by generating and submitting the best solution to an ETP instance using the VAST system. The one month competition allowed participants to spend as much time as they wished solving the problem, being able to log in and out of the system as required, resuming from their previously generated solution. Every action performed by participants was logged together with the generated solutions in a central database. As a further incentive to perform well a web page was maintained which displayed real-time scores for all of the competitors, calculated from the cost function of their current best solution. The following sections briefly summarise the results of this evaluation which are presented in full elsewhere[9].

5.1 Overall Performance

Table 3 summarises the overall performance of the top sixteen participants in the VAST evaluation. In total six participants generated feasible solutions with no hard constraint violations that could have been used with no further modification. Two of these six participants, spending around ten hours, generated solutions that were completely optimal in respect to the cost function used in the competition. The solutions generated by the remaining participants all had some conflict violations, or had failed to schedule all the exams, however many of the participants had generated good initial solutions that could have been refined to remove these hard constraint violations in a short period of time.

Table 3. Summary of the how many participants solved the problem, how long it took them and to what extent they solved the problem.

Description	Number of Participants	Mean Active Time
All Exams were assigned to the timetable with no conflict violations	6	6h 51m
Most exams were assigned and the solution may contain some conflicts. Some modification required.	5	2h 38m
Over an hour was spent solving the problem but it still contains many violations and significant work is required.	5	1h 29m

5.2 Strategies

It is important to show that the representation supports the use of different individual strategies rather than a biased perspective of the information that forces participants to adopt a particular strategy. When examining the strategies being used we considered the frequency of group assignments and reassignments compared to the manipulation of individual exams. Also number of exams in any group manipulations was examined together with the relative frequency of selection, assignment and refinement operations. The ordering these operations took place in and the impact of group manipulation on the required refinement operations was also looked at.

It was found that although there were differences in all these strategic components between participants no single strategies dominated or appeared significantly more effective than others. The participants who made the most use of the group exam selections and assignments appear to have benefited from being able to construct initial solutions relatively quickly, but the overall amount of time spent refining solutions appears to be the most critical feature of the more successful approaches.

Figure 7 shows one strategy, termed ongoing refinement, that was used to solve the problem. This figure clearly illustrates how the participant constructed a solution by selecting and assigning clusters of exams.

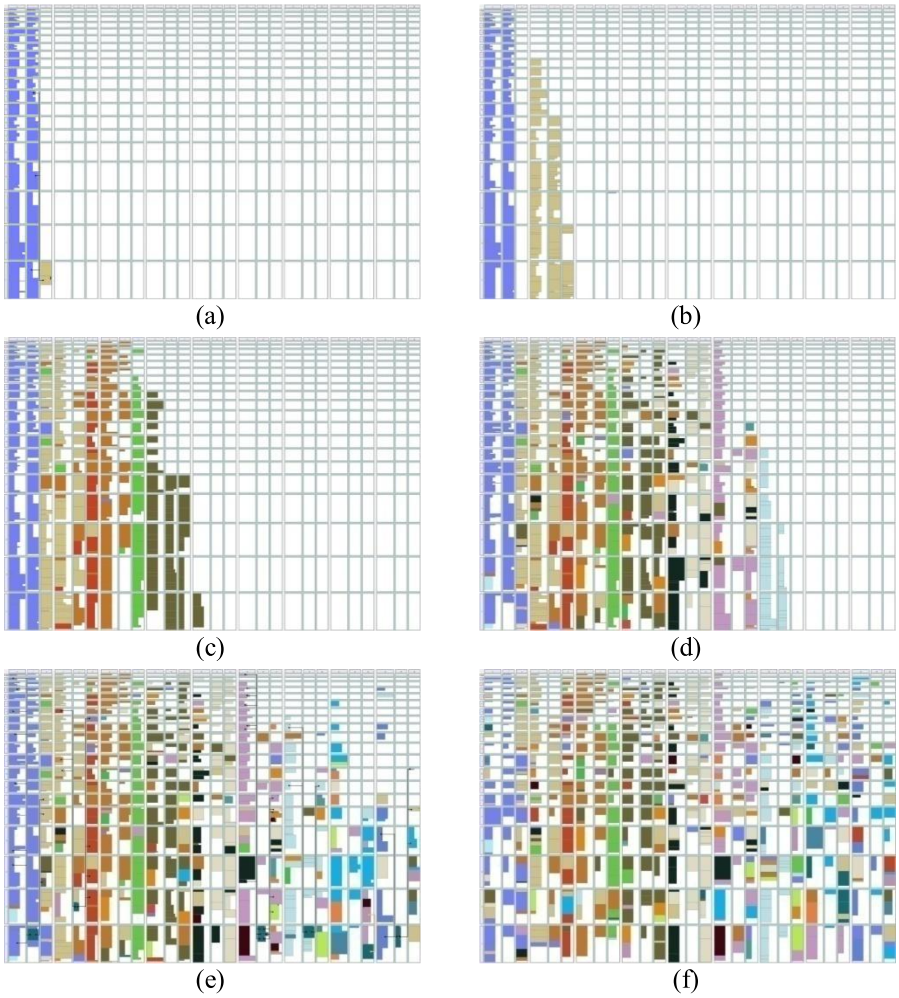


Fig. 7. Screenshots showing the progression of one participant toward solving the ETP instance. In this chronological series (Sub-figures a–e) a constructive approach can be seen where the clustering of exams from the Instance Visualisation has been used to initially assign the Exams to time periods from left to right over a period of around 50 minutes. Because of this assignment of clusters to time periods, and careful ongoing refinement, relatively few constraint violations appear in these solutions. Although some consecutive violations can be observed in Sub-figure (e) they have all been resolved in Su-Figure (f).

Previous analysis of REEP interfaces (e.g., [3]) has highlighted the use of recursion in human strategies and this was also discovered in this evaluation. This supports the claim that, rather than random trial and error strategies, in the most situations logical relationships exist between successive user actions. One example of this is forward shifting where assigning an exam to a time period creates a conflict violation and resolving this violation in turn creates another conflict violation and the cycle repeats.

6 Implications

In this paper we presented the application of the REEP-ID methodology to the generation of novel interfaces, interactive visualisations, to solving the Exam Timetabling Problem. The principles of REEP-ID were outlined together with the design of the VAST interface. In the previous section some of the results of an evaluation were presented indicating how effective VAST is in terms of generating complete solutions to the ETP.

Using the representational epistemic approach provides a methodology to design interfaces that effectively support the cognitive problem solving and perceptive abilities of humans. Integration of these visualisations within existing automated systems would greatly increase the levels of available interaction providing benefits such as the ability to perceive exactly what and where the major constraint violations are, and ways of interactively resolving these constraints or making other modifications without introducing new unforeseen problems.

The use of *glyphs* with this approach has generated a set of visual components which could be recombined and extended with new decorations to represent other similar timetabling problems with similar conceptual dimensions. Similarly institutions with unique constraints can introduce new decorations on these glyphs to represent their unique violations without breaking the unambiguous global cohesive framework of the visualisation or having to introduce entirely new representations or perspectives.

This research has provided further evidence to support the use of REEP design approaches to interface design for problem solving in these kinds of conceptually complex data rich domains. Whilst it is not envisaged that the VAST approach will ever replace automated solving systems these interactive visualisations do provide clear benefits to practitioners and suggest one possibility for improving interfaces and levels of interactivity and humanisation within this domain.

References

1. Cheng, P.C.-H. and R. Barone, *Representing complex problems: A representational epistemic approach*, in *Learning to solve complex scientific problems*, D.H. Jonassen, Editor. 2007, Lawrence Erlbaum Associates. p. 97-130.

2. Cheng, P.C.-H., *Electrifying diagrams for learning: Principles for effective representational systems.* Cognitive Science, 2002. **26**(6): p. 685-736.

3. Cheng, P.C.-H., et al., *Opening the Information Bottleneck in Complex Scheduling Problems with a Novel Representation: STARK Diagrams*, in *Diagrammatic representations and inference: Second International Conference*. 2002, Springer-Verlag: Berlin. p. 264 - 278.

4. Cheng, P.C.-H. and R. Barone. *Representing rosters: Conceptual Integration counteracts visual complexity*. in *Diagrammatic representation and*

inference: Third international conference, Diagrams 2004. 2004. Berlin: Springer-Verlag.

5. Kingston, J.H., *Modelling Timetabling Problems with STTL* in *Selected papers from the Third International Conference on Practice and Theory of Automated Timetabling III* 2001 Springer-Verlag. p. 309-321

6. Ranson, D. and S. Ahmadi, *An Extensible Modelling Framework for Timetabling Problems*, in *Practice and Theory of Automated Timetabling VI*, E.K. Burke and H. Rudova, Editors. 2007, Springer-Verlag. p. 383-393.

7. Burke, E., et al. *Examination Timetabling in British Universities - A Survey*. in *Practice and Theory of Automated Timetabling*. 1995. Napier University, Edinburgh: Springer-Verlag.

8. McCollum, B., et al. *A Review of Existing Interfaces of Automated Examination and Lecture Scheduling Systems(Abstract)*. in *Practice and Theory of Automated Timetabling*. 2002. KaHo St.-Lieven, Gent, Department of Industrial Engineering, Belgium.

9. Ranson, D., *Interactive Visualisations to Improve Automated Scheduling Systems*. DPhil thesis. 2008, University of Sussex.