Including Generative Mechanisms in Project scheduling using Hybrid Simulation

Jeff Scales

University of Technology Sydney, Australia
e-mail: jeff.scales@student.uts.edu.au

Keywords: project scheduling, hybrid simulation, system dynamics, discrete event simulation, multi-method modelling.

Abstract

Scheduling is central to the practice of project management and a topic of significant interest for the operations research and management science academic communities. However, a rigour-relevance gap has developed between the research and practice of scheduling that mirrors similar concerns current in management science. Closing this gap requires a more accommodative philosophy that can integrate both hard and soft factors in the construction of project schedules. This paper outlines one interpretation of how this can be achieved through the combination of discrete event simulation for schedule construction and system dynamics for variable resource productivity. An implementation was built in a readily available modelling environment and its scheduling capabilities tested. They compare well with published results for commercial project scheduling packages. The use of system dynamics in schedule construction allows for the inclusion of generative mechanisms, models that describe the process by which some observed phenomenon is produced. They are powerful tools for answering questions about *why* things happen the way they do, a type of question very relevant to practice.

Introduction

Scheduling, the construction and management of a rational time-based plan, is central to the practice of project management. The sixth edition of the Guide To The Project Management Body Of Knowledge (PMI 2017) recognises this more explicitly than previous editions, changing of name of the relevant knowledge area from time management to schedule management, more accurately describing the skill set the competent project manager is expected to master. The Project Management Institute (PMI), the premier standards setting institute for project management standards and practices, further emphasises the importance of scheduling by also publishing a practice standard for scheduling (PMI 2011). This distinguishes between scheduling methods, scheduling tools and the schedule model. Methods describe ways of constructing schedules, tools are usually software packages that embody the rules defining methods, and schedule models are created when methods are applied, using tools, to the specific data describing a project.

Scheduling is also a topic of significant interest for the operations research and management science academic communities. Research has focused mainly on scheduling methods; procedures for establishing the efficient allocation of time and resources to a set of activities, described generally as the Resource Constrained Project Scheduling Problem (RCPSP). Calculating an exact and optimal solution for a single project instance of the RCPSP is NP-hard (Demeulemeester & Herroelen 2002, pp. 203-5) and the challenge has attracted much interest (Hartmann & Briskorn 2010), requiring the RCPSP research community to establish a common language, borrowed from mathematics, to efficiently describe the problem (Brucker et al. 1999; Herroelen, Demeulemeester & De Reyck 2001). The community recognises however, that few of the published exact or metaheuristic scheduling methods have been used in practice (Herroelen 2005).

This is evidence of a gap between the research and practice of scheduling. Scheduling methods supported by the tools that practitioners rely on, do not attempt to obtain exact or optimal solutions to the RCPSP, instead settling for computationally less intensive solutions using methods based on heuristics (Trautmann & Baumann 2009). Heuristics are relatively simple rules for deciding how constrained resources should be allocated to the various activities requiring them (Artigues & Rivreau 2008; Demeulemeester & Herroelen 2002, pp. 264-300). Their simplicity allows them to be applied to the large precedence networks that occur in practice (+500 tasks) which would often result in unacceptable computing time if approached using an exact solution procedure. Not being exact solutions, means that schedules constructed using heuristics, cannot be guaranteed to be optimal. Results from scheduling tools popular with practitioners vary significantly in how closely they approach known optimal solutions, depending on the type of heuristic applied, whether more than one heuristic is used and the network and resource characteristics of the precedence network tested (Kastor & Sirakoulis 2009; Trautmann & Baumann 2009).

Further widening this gap is the fact that new scheduling challenges are continually being presented, as our understanding of the complexity of the project environment evolves. In his 2003 review of the contribution that mathematical modelling had made to the practice of project management, spanning the previous fifty years and dealing extensively with the RCPSP, Williams highlighted the inclusion of systemic and dynamic effects as particularly important. He noted that some researchers had begun to include such concepts in their planning assumptions, with the aim of explaining observed behaviours of projects based on feedback mechanisms and ‘soft factors’. These soft factors being described as qualitative, intangible or otherwise difficult to quantify concepts, that nevertheless had modelling utility (Williams 2003).

Soft factors also form a significant line of enquiry in the Rethinking Project Management literature (Svejvig & Andersen 2015; Walker & Lloyd-Walker 2016). One view sustained within this literature is that of the project as a complex dynamic system, whose behaviour is modified by rework cycles (Jalili & Ford 2016), the social nature of projects (Small & Walker 2010), human reactions to stress and burnout (Pinto, Patanakul & Pinto 2016) and myriad other, hard to quantify but observable effects.

The RCPSP community has already begun including one such soft factor as a planning assumption; learning curves. These describe assimilation rates for new skills and can be formulated in various ways (Anzanello & Fogliatto 2011). Learning curves have been included in scheduling methods based on linear programming (Mályusz & Varga 2017) design structure matrices (Huang & Chen 2006) and discrete event simulation (Yang et al. 2014). Research into scheduling with learning curve effects has been sufficiently broad in approach to prompt more than one state of the art review (Azzouz, Ennigrou & Ben Said 2017; Biskup 2008).

Soft factors descriptive of other interesting phenomena are also being included in formulations for scheduling problems other than single projects. These include; variable resource productivity in flow shop scheduling (Benavides, Ritt & Miralles 2014), models of fatigue for nurses in shift scheduling (Lin et al. 2013), for air traffic controllers (Wang & Ke 2013) and express delivery services (Lau, Woo & Choi 2006), and workforce scheduling with multitasking (Zhu, Li & Chu 2017) and skills and personality attributes (Othman, Gouw & Bhuiyan 2012). The increasing diversity in applications and approaches has even prompted a proposed taxonomy for integrating scheduling theory and soft factors (Lodree Jr, Geiger & Jiang 2009).

The inclusion of soft factors in project scheduling using so many different formalisms adds to the already considerable diversity evident in approaches to the RCPSP (Hartmann & Briskorn 2010). This diversity, whilst being indicative of a vibrant research community, may not always be a positive force, particularly if we espouse normative aims for our research. If we intend to inform practice we must remember who our audience is, or rather who they have been taught to become through their training in project management, with its emphasis on positivist epistemology, deductive reasoning and quantitative or reductionist techniques (Pollack 2007; Shepherd & Atkinson 2012). Through the positivist lens, which favours generalizable solutions, diversity can look like case-based research, where the relevance of findings is restricted to environments with similar attributes. When novel methods are operationalised using unfamiliar tools and the results presented in nonstandard formats it is difficult for the practitioner to recognise where relevant similarities exist.

Similar concerns have also been current in the broader discipline of management science and have been characterised as the rigour-relevance debate (Starkey & Madan 2001). This debate sees the ontological and epistemological differences between the dominant philosophical paradigms, the empirical-analytical, the interpretive and the critical, as being the source of the rigour-relevance gap. Bridging the gap requires the acceptance of a more accommodative philosophical framework and Mingers has been a strong proponent of Critical Realism (Mingers 2000, 2003, 2006a, 2011, 2015).

Critical Realism is a transcendental, realist, naturalist and critical philosophy of science (Bhaskar 1993, 2014; Bhaskar & Hartwig 2008) and a proper treatment of the argument for its usefulness in bridging the rigour-relevance gap is beyond the scope of a single paper on project scheduling. The element abstracted for this paper is the concept of the generative mechanism which Critical Realism posits resides in the domain of the real, has the power to create events in the domain of the actual which may or may not be observed in the domain of the empirical (Mingers 2006b).

Generative mechanisms are explanatory mechanisms; they describe the process by which an observed phenomenon is produced. Explanatory mechanisms have been contrasted with causal mechanisms in the philosophy of science for their utility in furthering research (Gerring 2007; Glennan 2002; Machamer 2004). Explanatory mechanisms can be physical explanations of how some observed phenomenon is generated or they can be epistemic explanations where the observation is understood within the context of what is already known (Illari & Williamson 2013; Illari & Williamson 2011; Williamson 2013).

The RCPSP research community needs to continue developing new and better scheduling methods, expanding the range of factors included as planning assumptions, but also needs to close the gap between research and practice. This paper proposes a method by which generative mechanisms utilising epistemic explanations of observable and potentially intangible factors can be included in project schedule construction. This will allow rigour to be maintained whilst potentially increasing relevance to practice by expanding the range of concepts used in the construction of project schedules.

The proposal includes two important assumptions. Firstly, that soft factors are best described as continuous rather than discrete variables. And secondly, that replicability and reproducibility require an open source approach to the tools used to demonstrate new methods (Kendall et al. 2016).

Describing soft factors as continuous variables allows the adoption of system dynamics as the formal system for modelling their behaviour. The system dynamics view that “the structure of a system gives rise to its behaviour” (Sterman 2000, p. 28) corresponds well with generative mechanisms in Critical Realism’s domain of the real (Mingers 2000, p. 1264). Since quantitative modelling in system dynamics is a simulation approach, all other calculations also need to occur within a simulation environment, leading to the adoption of discrete event simulation for scheduling (Lu 2003). Simulation also leads to the adoption of heuristics-based scheduling since algebraic methods are incommensurable with the treatment of time in simulations. Combining system dynamics and discrete event simulation makes this a hybrid simulation approach which significantly constrains the options for implementation, bearing in mind the goal of replicability and reproducibility. The approach described below has been implemented in the Anylogic™ multimethod modelling environment , describes as “is the standard in multimethod modeling technology, delivering increased efficiency and less risk when tackling complex business challenges” on the developer’s website (AnyLogic 2019).

Literature Review

In the canon of project management knowledge, scheduling is categorised as a planning activity and planning forms a very large part of what a competent project manager is expected to know. The Guide to the Project Management Body of Knowledge (PMI 2017) for instance, lists 49 management processes, 24 of which are devoted to planning. 8 of these planning processes are directed towards establishing the detail of who does what, when. Research also shows that the preferred tool for summarising the detail resulting from this planning activity is the Gantt chart (Besner & Hobbs 2004; Jugdev et al. 2013).

In terms of scheduling methods, the three most popular project management bodies of knowledge (APM 2012; PMAJ 2005; PMI 2017) focus almost exclusively on Critical Path Method (CPM) and leave the management of resource constraints to the commercially available software packages. Whilst these packages are also capable of Program Evaluation and Review Technique (PERT), stochastic scheduling remains less popular, in part due to ongoing concern over its assumption of a beta distribution (Trietsch et al. 2012; Vanhoucke, Coelho & Batselier 2016).

The popular commercial project scheduling packages all use heuristics-based scheduling methods for constraining resource usage and their capabilities have been the subject of periodic research. Comparison of the scheduling and resource allocation capabilities of the, often proprietary, methods embedded in commercial software, has required the invention of the capability to generate standardised project network data for the construction of test cases (Kolisch & Sprecher 1995) and the publishing of these test cases for use by the RCPSP community (Kolisch & Sprecher 1997). This has enabled comparison to be made using either variation from the unconstrained minimal makespan (Baumann & Trautmann 2016; Kolisch 1999) or from the growing database of published optimal solutions for test cases (Trautmann & Baumann 2009; TUM 2019) as well as data from real projects (Kastor & Sirakoulis 2009). Makespan is the technical term for the overall duration between the first and last tasks in a schedule.

The majority of research effort however, seems to have been focused on the RCPSP as an exercise in solution optimisation, with variants involving different formulations, or constraint relaxations (Artigues 2008; Demeulemeester & Herroelen 2002). This intensely mathematical approach has drawn criticism. Williams for instance observed that much of this research "languishes in journals" rather than finding use in project practice (Williams 2003, p. 3) and Herroelen similarly observed that "many project scheduling procedures have not yet found their way to practical use" (Herroelen 2005, p. 413). Herroelen's approach was to construct a hierarchical project planning and control framework to help close the rigour-relevance gap, under which, this research would be described as operational resource capacity planning. Williams, in contrast, advocated drawing on the view of projects as dynamic systems, a perspective he has had a long engagement with (Williams 1999, 2005; Williams et al. 1995).

Saaty too thought that operations research should be more than just 'optimisation subject to constraints' and that the community "would make a vast creative leap if it were to look seriously into models that handle intangibles and their measurement, because most of our problems deal with such factors" (Saaty 1998, p. 13). Such views have been current in the systems perspective for much longer. Forrester, the founder of system dynamics advised against "the omission of admittedly highly significant factors (most of the 'intangible' influences on decisions) because these are unmeasured or unmeasurable. To omit such variables is equivalent to saying they have zero effect … probably the only value that is known to be wrong" (Forrester 1961, p. 57).

Forrester's discipline of system dynamics has a long history of building generative mechanisms for observed project behaviour using causal loop and stock & flow diagrams that combine both tangible and intangible concepts (Love, Park & Han 2013; Sterman 2007). Some have been expanded into simulation models to animate this generation of behaviour and show how such concepts can interact, over time, in complex and dynamic ways (Park 2001). This analysis of the dynamics of projects has been considered "one of the most successful areas for the application of system dynamics" (Lyneis & Ford 2007, p. 157) and represents a rich source of material that has yet to be applied to project scheduling.

The project management community has made little use of this research. Rumeser and Emsley investigated the challenges of using system dynamics in project management. Their top five issues were; project manager's belief that they were already doing the right thing, the perception of system dynamics as 'scary', the perception of models as 'one-time solutions', the perception that the approach was not practical and the prelevence of attribution error regarding cause-effect relationships in the project environment (Rumeser & Emsley 2016).

The use of system dynamics for planning, rather than post hoc analysis remains rare, with a focus on construction projects (Hany et al. 2018; Lee, Peña-Mora & Park 2006; Peña-Mora & Li 2001). Within this research, system dynamics has also been combined with discrete event simulation, the former for modelling feedback effects from management decision making and the latter operational construction processes (Lee, Han & Peña-Mora 2007; Moradi, Nasirzadeh & Golkhoo 2015; Peña-Mora et al. 2008).

The technical difficulties involved in combining these two modelling approaches are a significant barrier to new entrants (Alzraiee, Moselhi & Zayed 2012; Alzraiee, Zayed & Moselhi 2012). However, commercial modelling software seems to have solved many of these issues. It is now possible to combine system dynamics, discrete event simulation and agent based modelling in a single off-the-shelf software tool such as Anylogic™ which is based on the widely used Java programming language (Borshchev 2013).

Whether the combination of system dynamics and discrete event simulation is described as multi-method, mixed-method, or hybrid modelling, researchers already using this approach in other fields believe it should be considered “no big deal” (Pidd 2012). An extended analysis has been made, focusing on examples from healthcare, of the practical and philosophical difficulties of combining these two modelling approaches, with the conclusion that whilst technically challenging they are not paradigmatically incommensurable (Brailsford, Churilov & Dangerfield 2014; Morgan, Howick & Belton 2017; Rotaru, Churilov & Flitman 2014).

System dynamics models of projects however, including those referenced above are generally high-level descriptions which aggregate all project tasks into a single smooth flow that best fits the calculus-based mathematics underlying stock & flow structures. This, of course, eliminates the very detail that resource constrained project scheduling seeks to manage. The system dynamics community has long recognised that “for closer examination of scheduling rules, a different modelling technique would be required, which would allow consideration of discrete events, rather than an aggregate stream of activity flow” (Kelly 1970, p. 18). This other modelling technique can be found in the work of Lu who has specifically investigated resource constrained critical path project scheduling (Lu & Lam 2008; Lu & Li 2003) based on a simplified discrete event simulation approach (Lu 2003).

Generative mechanisms are powerful tools for answering questions about *why* things happen the way they do, a type of question very relevant to practice. Rigour however, requires that we build such mechanisms from solid foundations. Discrete event simulation has a solid foundation for scheduling. System dynamics has a solid foundation for modelling both hard and soft factors as mechanisms which can include feedback loops. To utilise system dynamics in scheduling though we need a new stock & flow construction that is not based on an aggregation of tasks but nevertheless represents some commodity that is common to all the elements that make up a project schedule comprised of many separate and heterogeneous tasks. This stock & flow structure then needs to be embedded within a discrete event simulation-based scheduling engine making the combination a hybrid-simulation approach.

Stocks and Flows

Consider the view that project schedules represent time-based maps of transformations. Tasks scheduled at any moment in time are intended to be undergoing a transformation from a *to-do* state to a *done* state, with this transformation generating something that has utility for successor activities.

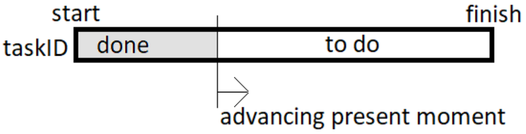

Figure 1 Progress represented using a time bar

The transformation ongoing at any moment can also be represented as a flow, with *to-do* and *done* being stocks. The *to-do* value of an activity might be expressed in dollars, as is done in earned-value analysis, or as *work-effort*, typically person-days.

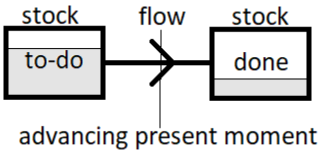

Figure 2 Progress represented using stock and flow

The flow here represents the expected completion rate for the task, based on the assumption that one person-day of estimated *work-effort* takes one person, one day to complete. This assumption is based on the understanding that part of the project management process assigns resources to tasks, based on skills, and the adjusts planned durations to match the work completion rate of the assigned resource's skill level. The matching of estimated *work-effort*, skill level, and duration has the effect of normalising the expected completion rate to unity. One person is expected to complete 3 person-days of effort at a rate of 1 per day over 3 days. This is a statement of the assigned person's *productivity*.

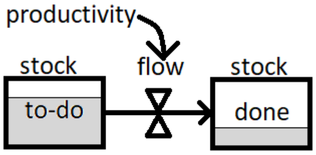

Figure 3 Productivity controls progress flow

Productivity governs the rate of flow of work *to-do* into work *done*, and the planning process normalises the expected *productivity* to unity, by adjusting expected duration, based on work content and the skill level of the assigned resources. This principle is expressed in the concept of the full-time equivalent resource and is used to build project plans prior to the assignment of named individuals. A full-time equivalent resource has unit *productivity* and achieves a work rate of 1 person-day of effort per day. In this view, stocks represent *work-effort* measured in person-days. Flows represent work achievement rates, measured in fractions of a person-day per time-period, and *productivity* is a controlling variable with a nominal value of 1.0.

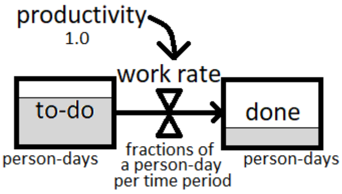

Figure 4 Stock and flow units

This structure represents first order flow where work *to-do* is drained, or work *done* accumulates at a rate defined only by *productivity*'s effect on *work-rate*. An important consequence of first order flow is that if separate amounts of *work-effort* are added to the stock of work *to-do*, their individuality is lost as they flow through to work *done*. This constraint but can be accommodated by making a simplifying assumption common in scheduling that resources do not multi-task, they only ever have one active assignment and must complete this before they are available for another. The simple stock & flow structure above includes the basic elements necessary for using system dynamics to calculate the *finish-time* for a single resource's allocated *work-effort* associated with a single task assignment. In the base case, where *productivity* is a constant 1.0 and *work-rate* represents that achieved by a full-time equivalent resource, this stock & flow structure should produce comparable plans to standard algebraic methods.

Discrete Event Scheduling

The stock & flow structure above is only relevant to calculating work progress for individual resources once the activity has been started. The management of precedence relationships, resource allocation and the setting of activity start, and end times must still be accomplished. However, they must now occur within an environment capable of supporting the calculus underlying system dynamics so that progress can be represented by a flow of *work-effort*.

The use of calculus-based system dynamics, albeit computer-based numerical calculus, means that this approach utilises continuous time modelling. This involves the assumption of a modelling constraint, that of the advancing present moment, created through the continuous accumulation of *time-step*s within the model as time is advanced. Calculations made at the moment described as ‘now’ in model time can only affect variables in existence at that instant and cannot have effects in the model’s ‘past’ or ‘future’. The advancing present moment means that scheduling must be based on heuristics, predefined rules for decision making that refer only to data available at the time the decision needs to be made.

Lu’s Simplified Discrete-Event Simulation Approach (SDESA) demonstrates how heuristics based scheduling can be achieved using queue sorting within a discrete event simulation environment (Lu 2003). Lu originally proposed SDESA as a scheduling approach for cyclic networks, where tasks are repeated until some gating condition is fulfilled but the approach is also applicable to acyclic networks. Several concepts used in SDESA have been carried forward in designing an implementation that includes the stock & flow structures described above.

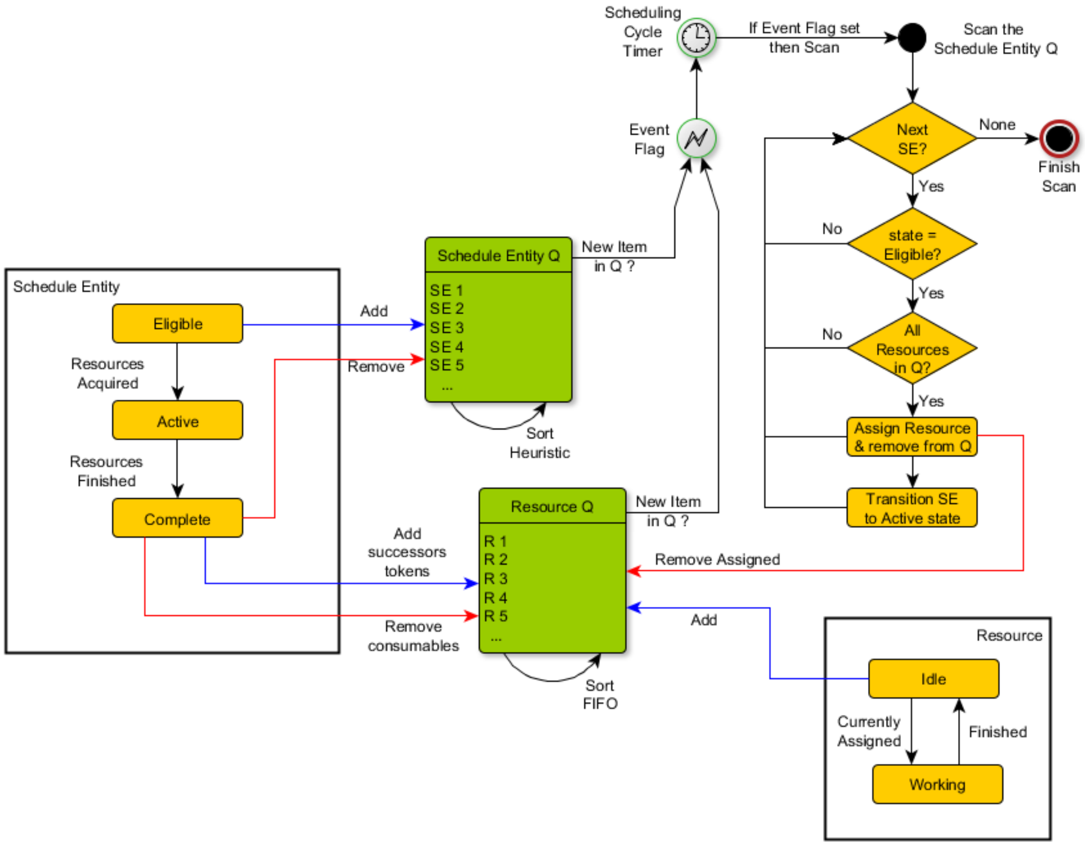

Figure 5 Overview of discrete event scheduling components

Two queues are used, one for resources (*ResourceQueue*) and another for schedule entities (*ScheduleEntityQueue*). To maintain the capability to support cyclic networks, a conceptual distinction is made between tasks and schedule entities. Task refers to the description of some package of work, which may or may not be performed more than once. Schedule entity refers to those instances where the task is planned to be performed, and therefore scheduled. In an acyclic network there will be a one-to-one relationship between tasks and schedule entities but in a cyclic network the same task may be repeated, creating multiple schedule entities. Schedule entities have states; *Pending*, *Eligible*, *Active* and *Complete* which are used in the same way as mutually exclusive sets in algebraic methods (Kolisch & Hartmann 1999).

Resources can be either renewable or consumable. Renewable resources return to the *ResourceQueue* on completion of their assignment whereas consumable resources are destroyed. Renewable resources also have states; *Idle* and *Working* (with sub-states *OnShift*, *OffShift* and *Finished*) which are also used in the scheduling process. A special type of consumable resource is a token created by a predecessor schedule entity for each of its successor schedule entities. Passing these tokens between entities is how network precedence relationships are maintained.

An executive process manages the allocation of resources to schedule entities and is initiated periodically to check whether an event has occurred which may allow a new entity to be scheduled. Such events occur whenever a new schedule entity or resource is added to or removed from either of their respective queues. As described below, schedule entities whose predecessors are all complete enter the *Eligible* state and on doing so add themselves to the *ScheduleEntityQueue*. Resources entering the *Idle* state add themselves to the *ResourceQueue*. Both events represent changes to the status quo and may present an opportunity to match a queued schedule entity to its required resources. The process is triggered by periodically checking an event flag rather than directly by events because the stock & flow structure controlling completion is not guaranteed to end at a discrete clock-time but will include some small stochastic component generated by the numerical calculus software. A truly greedy algorithm would attempt to schedule immediately at each event, giving the early arrivals a temporal priority, in addition to their actual priority under the scheduling heuristic being implemented. It was discovered during early testing that making this process periodic rather than event triggered, improved scheduling outcomes.

Both queues are maintained in a specific priority order. In this implementation the *ResourceQueue* is maintained in simple First-In-First-Out (FIFO) order, representing a time-in-queue priority rule for resource utilisation. The order of the *ScheduleEntityQueue* however, is used to implement various scheduling heuristics by sorting the queued entities based on some parameter value. Critical Path Method for instance would require schedule entities to have a parameter indicating float (slack) and sorting for ascending values of this parameter. After selecting the first queued entity in the sorted *ScheduleEntityQueue* the executive process then scans the *ResourceQueue* for the full complement of resources that this individual schedule entity requires. This implements a scheduling rule often described as ‘meeting mode’ where activities require the availability of all resources rather than some lesser quorum number. If all necessary resources are in the queue, they are assigned to the schedule entity, removed from the *ResourceQueue* and the stock & flow structure for each individual resource is initiated. If the full complement of required resources is not available, the executive process moves on the next entity in the sorted *ScheduleEntityQueue*.

Schedule entities can be in one of four states. The initial state is *Pending*, indicating that the entity is not yet involved in the scheduling process. When all predecessors for a schedule entity are complete, indicated by the availability of a full set of predecessor tokens, the schedule entity enters the *Eligible* state. It is then added to the *ScheduleEntityQueue* and contends for resources. Schedule entities which have secured all their resources enter the *Active* state and wait for communication from their assigned resources before setting their *begin-time*. This ensures that the *begin-time* includes any constraints arising from resource calendars. Schedule entities remain in the *Active* state until their full complement of assigned resources signal that they have completed their allocated *work-effort*, which is controlled by their stock & flow structure and their calendar.

Once all assigned resources have individually indicated completion, the schedule entity enters the *Complete* state and set its *finish-time*. It also performs some housekeeping activities on the *ResourceQueue* such as removing resources that have been consumed and adding tokens for its successors to communicate its own completion.

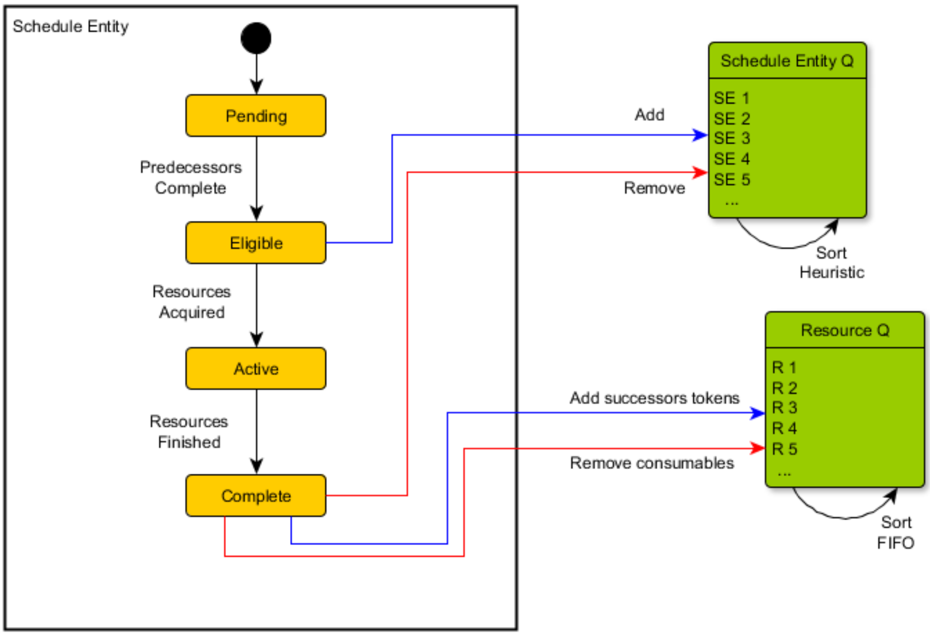

Figure 6 Schedule Entity state diagram

Resources have two main states, *Idle* and *Working*, dependent on whether they are currently assigned to a schedule entity or not. The *Working* state has three further sub-states; *OffShift*, *OnShift* and *Finished*. On entering the *Working* state, the value of *work-effort* that the resource is required to perform is added to the stock *ToDo*, but progress only takes place in the *OnShift* sub-state. *OnShift* represents calendar working time and gates the flow in the stock & flow structure simulating progress. *OffShift* represents calendar non-working time, night-time and weekends in a standard working week. Transitions between *OnShift* and *OffShift* are controlled by the calendar. On entering the *OnShift* state for the first time on a new assignment, resources signal their assigned schedule entity, allowing it to set it's *begin-time*. Progress on the resource's assigned *work-effort* is then simulated by the interaction of the resource's calendar effecting *OffShift* to *OnShift* state changes and *productivity* modulating the *work-rate* draining the *ToDo* stock into the *Done* stock.

Once enough progress has occurred to drain the *ToDo* stock to some defined threshold level (*residual threshold*) the resource transitions to the *Finished* sub-state and signals its assigned schedule entity that this individual resource has competed its assigned *work-effort*. The resource then enters the *Idle* state and is added back to the *ResourceQueue*, making it available for other queued schedule entities. The *residual threshold* is required because the numerical calculus underlying the stock & flow structure is iterative and will continue so long as the result can be represented numerically. In scheduling we are concerned with time on the human scale, down to hours and minutes, so stock levels representing amounts smaller than this can be considered residuals and ignored. Setting a value for the *residual threshold* allows this recursive process to exit when the required precision is achieved.

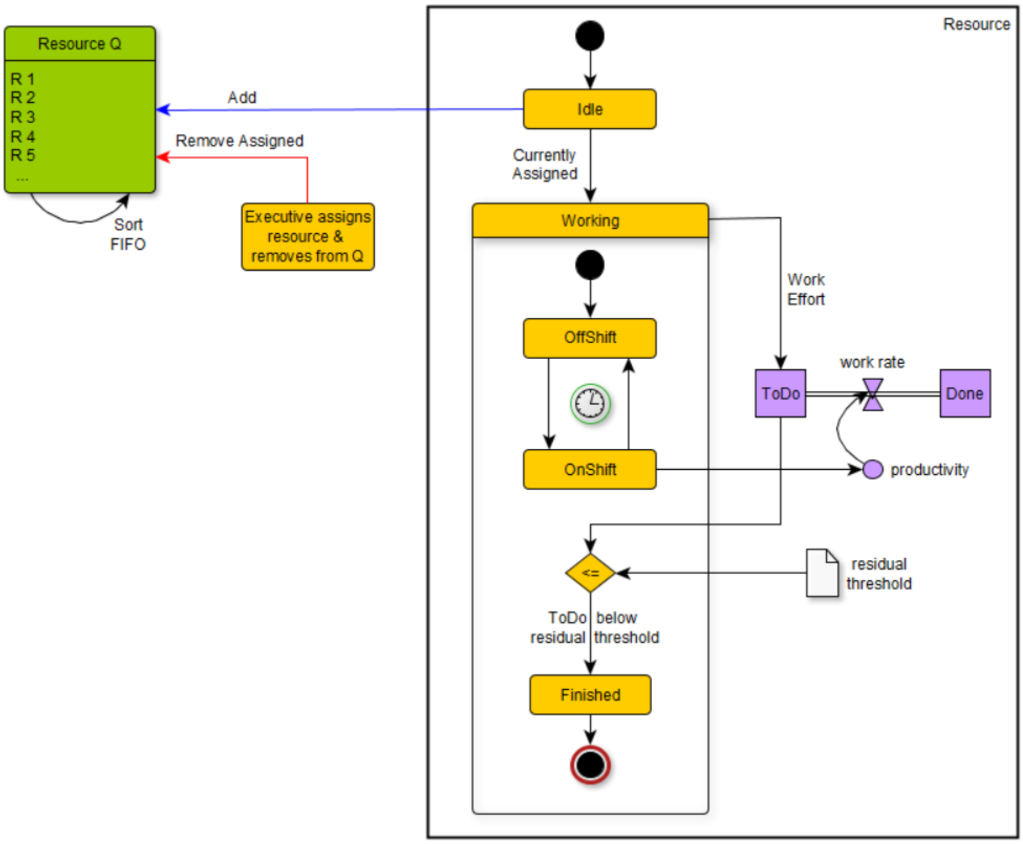

Figure 7 Resource state diagram

Since schedule entities in the *Pending* state are not queued, only the remaining states need to be considered as contending for resources. Further limiting this consideration to schedule entities only in the *Eligible* state, implements a simplification common in both the academic and practice communities, that of unary resource assignment. When applying this simplification, resources are assumed to have only one assignment at any time and multi-tasking is not allowed, matching the first order flow constraint identified above.

Thus, given an acyclic network and sorting the *ScheduleEntityQueue* for minimum float (slack), only considering schedule entities in the *Eligible* state as contending for the available resources, then removing those assigned resources from the *ResourceQueue*, will generate a CPM schedule that is a member of the set of time and resource feasible strict order solutions implementing unary and meeting mode resource assignment.

Benchmarking

Simulation dictates a heuristics-based approach to resource constrained project scheduling and therefore cannot guarantee solution optimality. The scheduling capabilities of such an approach will vary, as do results from commercial project scheduling software packages, depending on the characteristics of the task network and the type and number of heuristics employed. To be able to assess the fitness of such an approach from a scheduling perspective, some standardised dataset, such as the Project Scheduling Problem Library (PSPLIB, TUM 2019), is required. The PSPLIB contains precedence networks of tasks generated by an algorithm to exhibit certain desired characteristics and is used by the RCPSP research community to benchmark their different approaches to building scheduling algorithms.

The task networks in the PSPLIB are designed to exhibit several useful characteristics. They are acyclic, so CPM values can be calculated. They are Activity-On-Node networks, rather than Precedence-Diagram-Networks, so relationships are restricted to finish-to-start and do not include lags. Durations and resources are described using natural numbers, so fractions of a resource need not be considered. Resource pre-emption is not allowed, meaning that resource assignments are unary. Resources can be consumable or renewable but are restricted to being of only four different types (Kolisch & Sprecher 1997). Datasets are available for both the single-mode and multi-mode resource assignment cases, where the latter considers alternative ways of getting a task done. The administrators of the PSPLIB also maintain a register of current best solutions, many of which are known to represent the optimal solution, i.e. the minimum possible makespan (overall duration) under the defined resource constraints.

Trautmann and Baumann used the 30, 60 and 120 task single-mode PSPLIB datasets to benchmark the resource constrained makespan minimisation capabilities of seven commercially available project scheduling software packages using their default scheduling heuristic(s). They ranked them in various ways based on relative deviation from the current best solution (Trautmann & Baumann 2009). They later revisited the issue, testing eight packages against just the 120 task single-mode dataset but this time benchmarking each for relative deviation from the critical path lower bound, using various combinations of heuristics (Baumann & Trautmann 2016).

Results

An implementation of the simulation-based scheduling approach described above was created in the Anylogic™ modelling environment. It includes the capability to display results in Gantt chart form along with resource utilisation charts as these are the most common presentation styles of RCPSP instance data. CPM and other parameter data were calculated for all tasks in a network as the basis for simple numerical priority scheduling heuristics. These values were calculated once, prior to simulation, and were not updated as the simulation progressed. Eight simple numerical priority heuristics were tested. From CPM; minimum total float, minimum free float, early start and minimum late finish. Also included were minimum and maximum work content, a measure of how much effort a task represents and minimum and maximum duration, a measure of how short or long a task is.

Tests were performed using the PSPLIB 30 task (n=480), 60 task (n=480) and 120 task (n=600) single-mode datasets (total n=1560). Each of the eight simple heuristics were used to construct a schedule for each of the test instances and the makespan recorded. This makespan was then compared to the best known solution published on the PSPLIB website for that instance (TUM 2019) and a difference calculated. This difference was then expressed as a percentage variation from this best known figure, representing the quality of the schedule constructed by that heuristic, for that instance, from the perspectice of makespan minimisation.

$$Relative\ Makespan\ Deviation\ \% = \left(\frac{Simulation\ Result - PSPLIB\ Best}{PSPLIB\ Best}\right) \times 100$$

Simple statistical analyses were then performed on the 30, 60 and 120 task datasets of the Relative Makespan Deviation % to enable direct comparison with results published in Trautmann & Baumann (2009). Table 1 below summerises the comparison of arithmetic mean data. The upper half of the table shows data taken from the first 3 columns in Table 2 in Trautmann & Baumann (2009) comprising their results (T&B 2009 Results) for the default heuristic tests from seven scheduling tools including Microsoft Project™ (MSP) and Primavera™ P6 (PP6), full names and versions of these packages are published in Table 1 of their paper. The lower half of the table shows results from the simulation benchmarking tests (Simulation Model Results).

Table 1 T&B Default Heuristic and Simulation Model Results, comparison of Mean Relative Makespan Deviation %

	PSPLIB Datasets	30 Tasks	60 Tasks	120 Tasks
Heuristic		**Mean Relative Makespan Deviation %**		
T & B 2009 Results	ACO default option	9.10	8.21	19.62
	ATP default option	5.67	5.72	12.55
	CSP default option	8.12	9.62	21.23
	MSP default option	5.18	6.51	15.19
	PP6 default option	9.45	10.53	24.14
	PS8 default option	4.93	5.25	12.26
	TPP default option	8.96	10.23	25.32
Simulation Model Results	Min Total Float	6.57	7.21	16.91
	Min Work Content	10.69	12.14	26.67
	Max Work Content	8.96	10.92	23.56
	Min Late Finish	**4.05**	**5.24**	**10.31**
	Early Start	8.17	9.62	20.31
	Min Duration	9.83	11.12	23.79
	Min Free Float	6.88	8.15	18.98
	Max Duration	9.79	11.29	24.98

The simple heuristic of minimum late finish has been highlighted in bold as this represents the best of the results from the simulation model. They also represent the lowest mean relative makespan deviation % in the datasets in this comparison.

Table 2 below shows a table of the variance in each result set. The upper half of the table shows data taken from Table 3 in Trautmann & Baumann (2009) comprising their results for the variance of the relative makespan deviation % for their default heuristic tests. The lower half of the table shows the model results. Minimum late finish is again highlighted and represents the lowest variance of all the results in this comparison.

Table 2 T&B Default Heuristic and Simulation Model Results comparison of Variance of Relative Makespan Deviation %

	PSPLIB Datasets	30 Tasks	60 Tasks	120 Tasks
Heuristic		**Variance of Relative Makespan Dev**		
T & B 2009 Results	ACO default option	122.78	128.50	113.52
	ATP default option	56.17	71.84	79.89
	CSP default option	81.71	99.04	74.89
	MSP default option	44.66	68.88	102.71
	PP6 default option	93.43	120.41	99.22
	PS8 default option	48.52	62.83	68.23
	TPP default option	118.71	168.58	235.70
Simulation Model Results	Min Total Float	75.10	85.88	92.22
	Min Work Content	132.36	157.88	139.81
	Max Work Content	88.12	130.61	105.74
	Min Late Finish	**26.40**	**37.99**	**35.22**
	Early Start	65.57	81.62	57.14
	Min Duration	99.75	118.96	97.73
	Min Free Float	62.97	74.90	74.91
	Max Duration	117.50	142.31	113.40

Table 3 below shows a similar comparison for Trautmann & Baumann’s (2009) data for the mean relative makespan deviation % for the *best* single heuristic available in each of the seven packages they tested. Some of the commercial packages show the same result as in Table 1 as they have only one proprietary scheduling heuristic. The simulation model results for minimum late finish are again highlighted however, this time some of the commercial packages have better results with PP6 being consistently better across all three test datasets.

Table 3 T&B Best Heuristic and Simulation Model Results comparison of Mean Relative Makespan Deviation %

	PSPLIB Datasets	30 Tasks	60 Tasks	120 Tasks
Heuristic		**Mean Relative Makespan Deviation %**		
T & B 2009 Results	ACO best option	3.66	4.64	11.78
	ATP best option	5.67	5.72	12.55
	CSP best option	3.31	5.46	14.68
	MSP best option	5.18	6.51	15.19
	PP6 best option	2.38	3.75	9.89
	PS8 best option	4.93	5.25	12.26
	TPP best option	8.61	9.92	24.42
Simulation Model Results	Min Total Float	6.57	7.21	16.91
	Min Work Content	10.69	12.14	26.67
	Max Work Content	8.96	10.92	23.56
	Min Late Finish	**4.05**	**5.24**	**10.31**
	Early Start	8.17	9.62	20.31
	Min Duration	9.83	11.12	23.79
	Min Free Float	6.88	8.15	18.98
	Max Duration	9.79	11.29	24.98

Conclusions

This paper has taken the view that the gap between scheduling research and practice arises from the same ontological and epistemological issues that have been identified in the rigour-relevance debate within the broader management science community and that the solutions proposed there will be effective in project management too. This will entail taking a systems-based view of what generates the kind of behaviour observed on real projects and building generative mechanisms for this behaviour that can be included in schedule construction.

There are many ways in which such generative mechanisms might be included in the process of schedule construction. This paper has described one approach using discrete event modelling for scheduling and system dynamics for task progress. Abstracting task progress in this way and modelling it using system dynamics will hopefully enable the wealth of research that already exists incorporating the concept to be leveraged for the purposes of schedule construction.

Before that can happen though the approach must first prove its capabilities in building standard schedules, where task progress is nominal and constant. The data presented above shows that when compared with single heuristic scheduling in commercial packages the results can be good, dependent on the heuristic used. The commercial packages are of course quicker in calculating a result, the model requiring approximately one second per task to generate a schedule on the test PC (i7-6700, 3.40GHz, 4 Cores). Further tests of multiple nested heuristics are ongoing and can be compared in the same way with the existing published benchmark tests (Baumann & Trautmann 2016).

The model and benchmarking tests above represent an important proof of principle with several benefits evident in the implementation tested. The scheduling and generative mechanism components in the model are separate, mapping to those parts based on discrete event simulation and system dynamics respectively. Scheduling is implemented using standard Anylogic™ graphical components from the discrete event simulation palette plus additional code written in Java. Similarly, task progress is implemented using simple graphical components from the system dynamics palette. Deep knowledge of system dynamics is not required to conduct research into heuristics or meta-heuristics and deep knowledge of scheduling is not required to build generative mechanisms that modify resource productivity. The model provides a bridge between these two disciplines.

Closing the rigour-relevance gap will require researchers to accept some level of commonality in the three components of methods, tools and model presentation. The reuse of common components would serve to concentrate effort and allow the accumulation of comparable data on interesting phenomena from multiple sources, an approach likely to carry significant weight with practitioners. The ready availability of these components would allow interesting results to be replicated in their original context, or reproduced in new contexts, further supporting this accumulation of evidence. Component reuse would also ease the burden placed on researchers of acquiring new knowledge and skills as they attempt to push the boundaries of what is becoming an increasingly interdisciplinary area of research requiring good laboratory practice.

Acknowledgements

This research is supported by an Australian Government Research Training Program Scholarship. Supervisors; Prof. Shankar Sankaran, Dr Leila Moslemi Naeni.

References

Alzraiee, H., Moselhi, O. & Zayed, T. 2012, 'A hybrid framework for modeling construction operations using discrete event simulation and system dynamics', *Construction Research Congress 2012: Construction Challenges in a Flat World*, pp. 1063-73.

Alzraiee, H., Zayed, T. & Moselhi, O. 2012, 'Methodology for synchronizing Discrete Event Simulation and System Dynamics models', *Proceedings of the 2012 Winter Simulation Conference (WSC)*, pp. 1-11.

AnyLogic 2019, *Anylogic Multimethod Modelling Software*, The AnyLogic Company, viewed 20 May 2019 2019, <https://www.anylogic.com/>.

Anzanello, M.J. & Fogliatto, F.S. 2011, 'Learning curve models and applications: Literature review and research directions', *International Journal of Industrial Ergonomics*, vol. 41, no. 5, pp. 573-83.

APM 2012, *APM Body of Knowledge*, 6th edn, Association of Project Management, Princes Risborough.

Artigues, C. 2008, 'The Resource Constrained Project Scheduling Problem', *Resource-constrained project scheduling: models, algorithms, extensions and applications*, John Wiley & Sons, pp. 21-36.

Artigues, C. & Rivreau, D. 2008, 'Heuristics', *Resource-constrained project scheduling: models, algorithms, extensions and applications*, John Wiley & Sons, pp. 87-105.

Azzouz, A., Ennigrou, M. & Ben Said, L. 2017, 'Scheduling problems under learning effects: classification and cartography', *International Journal of Production Research*, pp. 1-20.

Baumann, P. & Trautmann, N. 2016, 'A note on the selection of priority rules in software packages for project management', *Flexible Services and Manufacturing Journal*, vol. 28, no. 4, pp. 694-702.

Benavides, A.J., Ritt, M. & Miralles, C. 2014, 'Flow shop scheduling with heterogeneous workers', *European Journal of Operational Research*, vol. 237, no. 2, pp. 713-20.

Besner, C. & Hobbs, B. 2004, 'An empirical investigation of project management practice: in reality, which tools do practitioners use?', *Innovations: Project management research*, pp. 337-51.

Bhaskar, R. 1993, *Dialectic: The Pulse of Freedom*, Verso.

Bhaskar, R. 2014, *The possibility of naturalism: A philosophical critique of the contemporary human sciences*, Routledge.

Bhaskar, R. & Hartwig, M. 2008, *A Realist Theory of Science*, Routledge, London.

Biskup, D. 2008, 'A state-of-the-art review on scheduling with learning effects', *European Journal of Operational Research*, vol. 188, no. 2, pp. 315-29.

Borshchev, A. 2013, *The big book of simulation modeling: multimethod modeling with AnyLogic 6*, AnyLogic North America.

Brailsford, S., Churilov, L. & Dangerfield, B. 2014, *Discrete-Event Simulation and System Dynamics for Management Decision Making*, John Wiley & Sons.

Brucker, P., Drexl, A., Möhring, R., Neumann, K. & Pesch, E. 1999, 'Resource-constrained project scheduling: Notation, classification, models, and methods', *European Journal of Operational Research*, vol. 112, no. 1, pp. 3-41.

Demeulemeester, E.L. & Herroelen, W. 2002, *Project Scheduling : A Research Handbook*, Kluwer Academic Publishers, Secaucus, UNITED STATES.

Forrester, J.W. 1961, *Industrial dynamics*, Pegasus Communications:, Waltham, MA.

Gerring, J. 2007, 'The Mechanismic Worldview: Thinking Inside the Box', *British Journal of Political Science*, vol. 38, no. 1, pp. 161-79.

Glennan, S. 2002, 'Rethinking Mechanistic Explanation', *Philosophy of Science*, vol. 69, no. S3, pp. S342-S53.

Hany, L., Hesham, O., Maged, G. & Moheeb, E. 2018, 'System Dynamics Approach for Forecasting Performance of Construction Projects', *Journal of Management in Engineering*, vol. 34, no. 1, p. 04017049.

Hartmann, S. & Briskorn, D. 2010, 'A survey of variants and extensions of the resource-constrained project scheduling problem', *European Journal of Operational Research*, vol. 207, no. 1, pp. 1-14.

Herroelen, W. 2005, 'Project Scheduling—Theory and Practice', *Production & Operations Management*, vol. 14, no. 4, pp. 413-32.

Herroelen, W., Demeulemeester, E. & De Reyck, B. 2001, 'A note on the paper “Resource-constrained project scheduling: Notation, classification, models and methods” by Brucker et al', *European Journal of Operational Research*, vol. 128, no. 3, pp. 679-88.

Huang, E. & Chen, S.-J. 2006, 'Estimation of project completion time and factors analysis for concurrent engineering project management: a simulation approach', *Concurrent Engineering*, vol. 14, p. 329+.

Illari, P. & Williamson, J. 2013, 'In Defence of Activities', *Journal for General Philosophy of Science*, vol. 44, no. 1, pp. 69-83.

Illari, P.M. & Williamson, J. 2011, 'Mechanisms are real and local', in P. Illari, F. Russo & J. Williamson (eds), *Causality in the sciences*, Oxford University Press, Oxford, pp. 818 - 44.

Jalili, Y. & Ford, D.N. 2016, 'Quantifying the impacts of rework, schedule pressure, and ripple effect loops on project schedule performance', Other, vol. 32, John Wiley & Sons, Inc., pp. 82-96.

Jugdev, K., Perkins, D., Fortune, J., White, D. & Walker, D. 2013, 'An exploratory study of project success with tools, software and methods', *International Journal of Managing Projects in Business*, vol. 6, no. 3, pp. 534-51.

Kastor, A. & Sirakoulis, K. 2009, 'The effectiveness of resource levelling tools for Resource Constraint Project Scheduling Problem', *International Journal of Project Management*, vol. 27, no. 5, pp. 493-500.

Kelly, T.J. 1970, 'The Dynamics of R&D Project Management'.

Kendall, G., Bai, R., Błazewicz, J., De Causmaecker, P., Gendreau, M., John, R., Li, J., McCollum, B., Pesch, E., Qu, R., Sabar, N., Berghe, G.V. & Yee, A. 2016, 'Good Laboratory Practice for optimization research', *Journal of the Operational Research Society*, vol. 67, no. 4, pp. 676-89.

Kolisch, R. 1999, 'Resource Allocation Capabilities of Commercial Project Management Software Packages', *Interfaces*, vol. 29, no. 4, pp. 19-31.

Kolisch, R. & Hartmann, S. 1999, *Heuristic algorithms for the resource-constrained project scheduling problem: Classification and computational analysis*, Springer.

Kolisch, R. & Sprecher, A. 1995, 'Characterization and Generation of a General Class of Resource-Constrained Project Scheduling Problems', *Management Science*, vol. 41, no. 10, pp. 1693-704.

Kolisch, R. & Sprecher, A. 1997, 'PSPLIB - A project scheduling problem library: OR Software - ORSEP Operations Research Software Exchange Program', *European Journal of Operational Research*, vol. 96, no. 1, pp. 205-16.

Lau, H.Y.K., Woo, S.O. & Choi, C.Y. 2006, 'Manpower allocation and shift scheduling using human performance simulation', *IFAC Proceedings Volumes*, vol. 39, no. 3, pp. 181-6.

Lee, S., Han, S. & Peña-Mora, F. 2007, 'Hybrid System Dynamics and Discrete Event Simulation for Construction Management', *Computing in Civil Engineering (2007)*.

Lee, S.H., Peña-Mora, F. & Park, M. 2006, 'Dynamic planning and control methodology for strategic and operational construction project management', *Automation in Construction*, vol. 15, no. 1, pp. 84-97.

Lin, R.-C., Sir, M.Y., Sisikoglu, E., Pasupathy, K. & Steege, L.M. 2013, 'Optimal nurse scheduling based on quantitative models of work-related fatigue', *IIE Transactions on Healthcare Systems Engineering*, vol. 3, no. 1, pp. 23-38.

Lodree Jr, E.J., Geiger, C.D. & Jiang, X. 2009, 'Taxonomy for integrating scheduling theory and human factors: Review and research opportunities', *International Journal of Industrial Ergonomics*, vol. 39, no. 1, pp. 39-51.

Love, P.E.D., Park, M. & Han, S. 2013, 'System Dynamics Modeling in the Project Environment', *Mathematical and Computer Modelling*, vol. 57, no. 9, pp. 2029-31.

Lu, M. 2003, 'Simplified Discrete-Event Simulation Approach for Construction Simulation', *Journal of Construction Engineering and Management*, vol. 129, no. 5, pp. 537-46.

Lu, M. & Lam, H.-C. 2008, 'Critical Path Scheduling under Resource Calendar Constraints', *Journal of Construction Engineering and Management*, vol. 134, no. 1, pp. 25-31.

Lu, M. & Li, H. 2003, 'Resource-Activity Critical-Path Method for Construction Planning', *Journal of Construction Engineering and Management*, vol. 129, no. 4, pp. 412-20.

Lyneis, J.M. & Ford, D.N. 2007, 'System dynamics applied to project management: a survey, assessment, and directions for future research', *System Dynamics Review*, vol. 23, no. 2-3, pp. 157-89.

Machamer, P. 2004, 'Activities and Causation: The Metaphysics and Epistemology of Mechanisms', *International Studies in the Philosophy of Science*, vol. 18, no. 1, pp. 27-39.

Mályusz, L. & Varga, A. 2017, 'An Estimation of the Learning Curve Effect on Project Scheduling with Calendar Days Calculation', *Procedia Engineering*, vol. 196, pp. 730-7.

Mingers, J. 2000, 'The Contribution of Critical Realism as an Underpinning Philosophy for OR/MS and Systems', *The Journal of the Operational Research Society*, vol. 51, no. 11, pp. 1256-70.

Mingers, J. 2003, 'A Classification of the Philosophical Assumptions of Management Science Methods', *The Journal of the Operational Research Society*, vol. 54, no. 6, pp. 559-70.

Mingers, J. 2006a, 'A Critique of Statistical Modelling in Management Science from a Critical Realist Perspective: Its Role within Multimethodology', *The Journal of the Operational Research Society*, vol. 57, no. 2, pp. 202-19.

Mingers, J. 2006b, *Philosophical Foundations: Critical Realism*, Springer US, Boston, MA.

Mingers, J. 2011, 'The Contribution of Systemic Thought to Critical Realism', *Journal of Critical Realism*, vol. 10, no. 3, pp. 303-30.

Mingers, J. 2015, 'Helping business schools engage with real problems: The contribution of critical realism and systems thinking', *European Journal of Operational Research*, vol. 242, no. 1, pp. 316-31.

Moradi, S., Nasirzadeh, F. & Golkhoo, F. 2015, 'A hybrid SD–DES simulation approach to model construction projects', *Construction Innovation*, vol. 15, no. 1, pp. 66-83.

Morgan, J.S., Howick, S. & Belton, V. 2017, 'A toolkit of designs for mixing Discrete Event Simulation and System Dynamics', *European Journal of Operational Research*, vol. 257, no. 3, pp. 907-18.

Othman, M., Gouw, G.J. & Bhuiyan, N. 2012, 'Workforce scheduling: A new model incorporating human factors', *2012*, p. 26.

Park, M. 2001, 'Dynamic planning and control methodology for large-scale concurrent construction projects', PhD thesis, Massachusetts Institute of Technology.

Peña-Mora, F., Han, S., Lee, S. & Park, M. 2008, 'Strategic-Operational Construction Management: Hybrid System Dynamics and Discrete Event Approach', *Journal of Construction Engineering and Management*, vol. 134, no. 9, pp. 701-10.

Peña-Mora, F. & Li, M. 2001, 'Dynamic Planning and Control Methodology for Design/Build Fast-Track Construction Projects', *Journal of Construction Engineering and Management*, vol. 127, no. 1, pp. 1-17.

Pidd, M. 2012, 'Mixing other methods with simulation is no big deal', *Proceedings of the 2012 Winter Simulation Conference (WSC)*, pp. 1-7.

Pinto, J.K., Patanakul, P. & Pinto, M.B. 2016, 'Project Personnel, Job Demands, and Workplace Burnout: The Differential Effects of Job Title and Project Type', *IEEE Transactions on Engineering Management*, vol. 63, no. 1, pp. 91-100.

PMAJ 2005, *A Guidebook of Project & Program Management for Enterprise Innovation*, Project Management Association of Japan, Tokyo.

PMI 2011, *Practice Standard for Scheduling – Second edition*, Second edition. edn, Project Management Institute, Newtown Square, Pennsylvania.

PMI 2017, *A guide to the project management body of knowledge (PMBOK® guide) – Sixth edition*, Sixth edition. edn, Project Management Institute, Newtown Square, Pennsylvania.

Pollack, J. 2007, 'The changing paradigms of project management', *International Journal of Project Management*, vol. 25, no. 3, pp. 266-74.

Rotaru, K., Churilov, L. & Flitman, A. 2014, 'Philosophical positioning of discrete-event simulation and system dynamics as management science tools for process systems', *Discrete-Event Simulation and System Dynamics for Management Decision Making*, pp. 76-104.

Rumeser, D. & Emsley, M. 2016, 'Key Challenges of System Dynamics Implementation in Project Management', *Procedia - Social and Behavioral Sciences*, vol. 230, pp. 22-30.

Saaty, T.L. 1998, 'Reflections and projections on creativity in operations research and management science: A', *Operations Research*, vol. 46, no. 1, pp. 9-16.

Shepherd, M. & Atkinson, R. 2012, 'True Science and Project Management', *European Conference on Research Methodology for Business and Management Studies*, Academic Conferences International Limited, p. 359.

Small, J. & Walker, D. 2010, 'The emergent realities of project praxis in socially complex project environments', *International Journal of Managing Projects in Business*, vol. 3, no. 1, pp. 147-56.

Starkey, K. & Madan, P. 2001, 'Bridging the Relevance Gap: Aligning Stakeholders in the Future of Management Research', *British Journal of Management*, vol. 12, p. S3.

Sterman, J.D. 2007, 'Exploring the next great frontier: system dynamics at fifty', *System Dynamics Review (Wiley)*, vol. 23, no. 2/3, pp. 89-93.

Sterman, J.D.J.D. 2000, *Business dynamics: systems thinking and modeling for a complex world*.

Svejvig, P. & Andersen, P. 2015, 'Rethinking project management: A structured literature review with a critical look at the brave new world', *International Journal of Project Management*, vol. 33, no. 2, pp. 278-90.

Trautmann, N. & Baumann, P. 2009, 'Resource-allocation capabilities of commercial project management software: An experimental analysis', *Computers & Industrial Engineering, 2009. CIE 2009. International Conference on*, pp. 1143-8.

Trietsch, D., Mazmanyan, L., Gevorgyan, L. & Baker, K.R. 2012, 'Modeling activity times by the Parkinson distribution with a lognormal core: Theory and validation', *European Journal of Operational Research*, vol. 216, no. 2, pp. 386-96.

TUM 2019, *Project Scheduling Problem Library - PSPLIB* Technische Universität München, viewed 9 May 2019 2019, <http://www.om-db.wi.tum.de/psplib/>.

Vanhoucke, M., Coelho, J. & Batselier, J. 2016, 'An overview of project data for integrated project management and control', *Journal of Modern Project Management*, vol. 3, no. 2, pp. 6-21.

Walker, D. & Lloyd-Walker, B. 2016, 'Rethinking project management: Its influence on papers published in the international journal of managing projects in business', *International Journal of Managing Projects in Business*, vol. 9, no. 4, pp. 716-43.

Wang, T.-C. & Ke, G.-C. 2013, 'Fatigue Minimization Work Shift Scheduling for Air Traffic Controllers', *International Journal of Automation and Smart Technology*, vol. 3, no. 2, pp. 91-9.

Williams, T. 1999, 'Towards realism in network simulation', *Omega*, vol. 27, no. 3, pp. 305-14.

Williams, T. 2003, 'The contribution of mathematical modelling to the practice of project management', *IMA Journal of Management Mathematics*, vol. 14, no. 1, pp. 3-30.

Williams, T. 2005, 'Assessing and moving on from the dominant project management discourse in the light of project overruns', *IEEE Transactions on Engineering Management*, vol. 52, no. 4, pp. 497-508.

Williams, T., Eden, C., Ackermann, F. & Tait, A. 1995, 'The Effects of Design Changes and Delays on Project Costs', *The Journal of the Operational Research Society*, vol. 46, no. 7, pp. 809-18.

Williamson, J. 2013, 'How Can Causal Explanations Explain?', *Erkenntnis*, vol. 78, no. 2, pp. 257-75.

Yang, Q., Lu, T., Yao, T. & Zhang, B. 2014, 'The impact of uncertainty and ambiguity related to iteration and overlapping on schedule of product development projects', *International Journal of Project Management*, vol. 32, no. 5, pp. 827-37.

Zhu, Z., Li, J. & Chu, C. 2017, 'Multitasking Scheduling Problems with Deterioration Effect', *Mathematical Problems in Engineering*, vol. 2017, p. 10.