SAND2017-12563C

Extensions of a Simple Temporal Network Coordinating Emergent Knowledge Processes in a Collaborative System-of-Systems

[Anonymous]
Affiliation omitted for double-blind reviewing.
Email removed for double-blind reviewing.

Abstract

The Z Machine is the world's most powerful x-ray source, routinely delivering over 20 MA of electrical current to targets in support of US nuclear stockpile stewardship and in pursuit of inertial confinement fusion. The large-scale, multi-disciplinary nature of experiments ("shots") on the Z Machine requires resources and expertise from disparate organizations to execute Emergent Knowledge Processes with independent functions and management, forming a Collaborative System-of-Systems. A previous work identified the resulting significant challenges of distributed planning and coordinating a given experiment day, and described one potential approach to scheduling based on a Simple Temporal Network with only minimum times between activities defined. The present work extends that approach in two ways. First, a method is proposed to establish latest cutoff times for activity starts through the setting of a single operational goal to back-schedule all latest times of when activities might begin to achieve that goal (so that unlike the lower bounds which are physically possible intervals, the upper bounds reflect operationally required times if the goal is to be achieved). Second, the present work implements a real-time web-based software tool to enable more informed planning of "shot day" activities and presenting information relevant on shot day to aid as an enabling interface between workers among the varied groups involved in planning and execution. The resulting software product is a scheduling tool that displays windows of time during which each activity could and should begin. The software's initial results are evaluated, and future areas for improvement are discussed.

Introduction

"Thou goest thine, and I go mine – many ways we wend;
Many days, and many ways, Ending in one end."
-George MacDonald, *Phantastes*

The Z Machine (hereafter "Z") is the world's most powerful x-ray source, routinely delivering over 20 MA of electrical current to targets in support of various programs, including US nuclear stockpile stewardship and pursuit of inertial confinement fusion. A single experiment (or "shot") requires months of planning, design work, specialized hardware fabrication, and diagnostics configuration, all involving experts from a variety of specialized backgrounds such as plasma physics, hydrodynamics, dynamic material properties, laser technologies, atomic spectroscopy, neutron diagnostics, electrical engineering, mechanical engineering, and electro-mechanical controls, among others. Execution of a shot can often be achieved in one day; regular operation of Z on a daily basis requires specialists from all of the fields above as well as technicians and installers performing regular machine maintenance and configuration. These personnel are involved in activities ranging from operating heavy machinery to refurbishing equipment, performing routine mechanical and electrical work, and even underwater diving.

A previous work (Anonymous 2017) identified Z and its participants as a Collaborative System-of-Systems (SoS) (Maier 1998) replete with Emergent Knowledge Processes (EKPs) (Markus 2000), exhibiting independent management and operation, volunteer-like participation, and unpredictable and emergent arrangements of people and systems. All of these traits create significant challenges to planning and scheduling activities for a given Z experiment (which can take anywhere from half a day to multiple days), which in turn create significant challenges to coordination of the various participants involved in the experiment – especially as configurations are redefined and as new needs emerge over the experiment's preparation and execution. This latter emergent condition is common, given the nature of EKPs, and yet it is also the most likely to cause previously communicated planning and scheduling information to become obsolete, whether or not all participants are aware of the developments.

Motivation: Aiding coordination of EKPs with an information system

Despite these challenges posed to planning and scheduling, "encouraging cooperation" of participants remains the primary goal of the present work in keeping with Maier's (1998) architectural principles for an SoS; efficient coordination is one of the means by which the present work attempts to encourage cooperation. In the present case, efficient coordination is enabled through an information system intended to provide a higher-level understanding of the anticipated and actual temporal behavior for a given experiment, which respectively reflect two of Maier's (2005) research challenges for the study of an SoS: Upper Layer Description and Upper Layer Analysis. Rhodes and Ross (2010) identify some of the behavioral and perceptual factors at play when attempting to describe the complexity of systems like Z, as described in (Anonymous 2017): success can only be achieved when the temporal behavior of a Z experiment is captured and presented in a way that can account for the varying perceptions of what that behavior means for individual participants, ultimately encouraging cooperation from each individual participant. Similarly, (Markus et al. 2000) state that since "the mix of backgrounds and expertise brought to bear on emergent knowledge processes can differ each time the process is performed," a meta-requirement for information systems that support EKPs "…must meet the diverse and sometimes contradictory needs of different user groups." In addition, since "no one individual or group has a complete grasp of both the general and specific knowledge that applies," a supporting information system "…must incorporate the frameworks and perspectives of several different kinds of participants" (Markus et al. 2000).

Method

Anonymous (2017) discussed some of the challenges to both the feasibility and usefulness of probabilistic information about scheduled activities for a Z experiment, concluding that communicating earliest time estimates (ETEs) instead of "likely" (or any other probabilistic) time estimates is the most effective approach to wide-scale coordination of independent agents in environments with large exogenous uncertainties (such as EKPs). Mass transit systems (e.g., planes, trains, buses) engage in widespread application of this idea of communicating earliest times for more efficient coordination of independent participants. However, stakeholders and research readers will still inevitably raise the question: why not use triangle distributions, or beta distributions, or PERT, or [insert favorite probabilistic method here] to describe each activity in the Simple Temporal Network, and therefore communicate so much more forecasting information to participants?

An evaluation of ETEs vs. probabilistic time estimates in EKPs

While Simon (1992) nicely sums up the big picture of why probabilistic methods are not appropriate in this context ("the heart of the data problem for design is not forecasting but constructing alternative scenarios for the future"), Markus et al. (2000) offer several specific, relevant meta-requirements for information systems that support EKPs. These meta-requirements are presently described and used to evaluate the appropriateness of earliest times vs. probabilistic times for coordination of participants in EKPs.

Meta-requirement: Because "knowledge must be actionable in application," an information system supporting EKPs "must be directed at improving off-line behavior and must tie knowledge to concrete practical action" (Markus et al. 2000). The question then arises, how is concrete practical action enabled by a probabilistic estimate of an activity's scheduled time? How should (or will) a participant change their actions if they receive an estimate of 30% confidence as opposed to 60% confidence, for example? Keeping in mind that participants each have their own cognitive biases, independent management, and voluntary participation with other participants, can they be expected to respond consistently, much less uniformly, to such information? (In addition, in a Collaborative SoS, who could enforce uniform responses even if they were desired by some participants?) Providing earliest times, on the other hand, provides the concrete action of "checking in" – communication is simple in the modern age, and a quick "check-in" is all it takes for a participant to find out whether to start an activity or check-in at a future time. With an ETE, therefore, a participant can be consistently prepared to begin an activity when needed, rather than showing up late because it was "not probable" that they'd be needed. In addition, participants can be off-line as much as they like after receiving an ETE, since (ideally) the estimate will not be invalidated by any future updates to that ETE.

Meta-requirement: Because "it is not possible to know in advance who will be involved" in an EKP, a supporting information system "must employ terms, operations, and an interface that are usable by participants who are unknown in advance" (Markus 2000). Setting aside the obvious (and significant) problems with relying on probabilistic information about processes that include unknown participants, the fact that unknown participants may be present compounds the challenges of the previous meta-requirement. Not only would the unknown participants need to be able to use the existing information effectively,

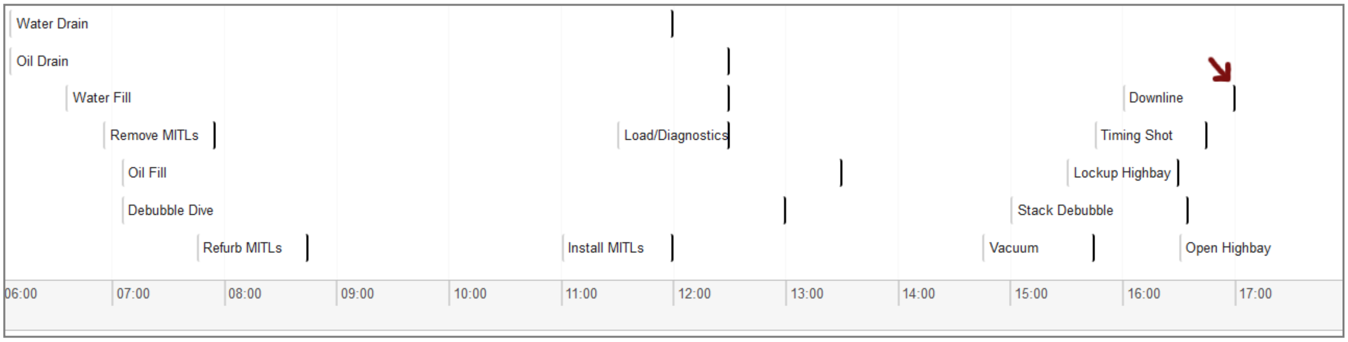

Figure 1. ETEs of activities (denoted by gray lines on each activity's left-hand side) are derived by scheduling the earliest times of activities starting at 0600. LCTs (black lines on right-hand sides) are derived by choosing a goal for Downline of 1700 (indicated by bold arrow), which allows minimum bounds to be back-propagated through the network to generate LCTs for all other activities.

but they would need to be able to provide their own probabilistic information so that the information system could update the probabilistic information about all following activities (likely invalidating the previously calculated values). It remains unclear how or when such information would be obtained from such participants (not to mention how such collection would be enforced), and it remains especially unclear how such information could be validated (since, as (Markus et al. 2000) point out, many participants "may participate only infrequently or in circumstances that do not reoccur"). ETEs, on the other hand, are agnostic to unknown participants, since the ETEs can always be pushed later in time without invalidating previously calculated ETEs of an activity or any of its successors.

Meta-requirement: Because participants in EKPs "may have considerable discretion over their use of methods and tools," a supporting information system must ensure "that decisions are easier to make with the system than without it." With probabilistic information, a participant could be confused, misled, or have other malformed cognitive functioning with respect to their actions in support of scheduled activities, making their real-time decisions potentially much more stressful and much more difficult to later analyze reflectively. In contrast, ETEs make decisions easier for every participant, since it is straightforward to infer when they might need to check-in in the future.

The meta-requirements above demonstrate the challenges to and ultimate insufficiency of providing probabilistic information to Z SoS participants as a means of effecting efficient coordination. They also demonstrate the appropriateness of providing ETEs in the context of Z's SoS, where EKPs are found throughout an experiment's lifecycle.

Providing Latest Cutoff Times (LCTs)

Anonymous (2017) raises the question of whether latest time estimates for activities could be provided in the same manner as ETEs, since Simple Temporal Networks naturally support both constructs. The meta-requirements above demonstrate that in addition to providing ETEs, providing latest time estimates for activities could also help participants (e.g., through enabling more appropriate off-line behavior and more concrete action). The SoS-style collaboration in EKPs in Z experiments, however, implies that upper bound estimates on experimental activities' durations are dubious at best, and misleading at worst. For many of the same reasons as with ETEs, probabilistic estimates cannot help here.

If lower bound temporal relationships between activities are established, however, then latest cutoff times (LCTs) for those activities could be derived through backward propagation of the lower bounds from some future "goal" time (i.e., back-scheduling). Since in the case of Z experiments all activities eventually lead to a Z Shot, an operational goal for the activity of shooting the machine can be de-fined (e.g., "Fire the machine by 5pm today"), which can be back-scheduled through the network to provide the latest cutoff times (LCTs) for all preceding activities[1]. An illustration of ETEs and LCTs can be seen in Figure 1.

Example derivation of ETE and LCT

Let G be the directed edge-weighted graph defined as a tuple **G** := <**V**, **E**>:

V: set of nodes, each representing the start of an activity (e.g., "Begin Water Fill")

E: set of edges ω_{ij} representing the minimum minutes between nodes, of form $j_{startTime} - i_{startTime} > \omega_{ij}$, where

$j, i \in \mathbf{V}$

i = source node for the directed edge

j = destination node for the directed edge

$\omega_{ij} \in \mathbb{R} > 0$

No cycles exist.

[1] The operational goal could also be mixed with other operational constraints (e.g., upper bounds of resource availability for individual activities) to incorporate more complete information on LCTs, but for present purposes only the shot event is considered.

Once **G** is constructed, the ETEs can be derived by scheduling all activities at their earliest times. To do so, the network is first ordered topologically using Kahn's (1962) algorithm, and then nodes without predecessors are assigned an ETE[2]. Successive nodes are then assigned ETEs in topological order through the calculation of:

$$ETE_j = \max \{ ETE_i + \omega_{ij}, \forall i, \omega_{ij} \in \boldsymbol{E}^-(j) \} \quad (1)$$

In other words, a node's ETE can be assigned by adding each incoming edge ω_{ij} (edge weight, in minutes) to that edge's source node's ETE, and taking the maximum (i.e., latest time) of this calculation for all incoming edges ($\boldsymbol{E}^-$).

Once the ETEs have been assigned, the latest cutoff time (LCT) for the goal node can be established (e.g., "5pm" for the Z Shot):

$$LCT_{ZShot} = 5pm$$

and then LCTs for all preceding activities can be assigned through back-propagation, or the inverse of the ETE calculations in Equation 1:

$$LCT_i = \min \{ LCT_j - \omega_{ij}, \forall j, \omega_{ij} \in \boldsymbol{E}^+(i) \} \quad (2)$$

The basic idea of Equation 2 is that a node's LCT can be assigned by calculating, for each of its outgoing edges ($\boldsymbol{E}^+$), the edge weight (in minutes) subtracted from the edge's destination node's LCT; the minimum time calculated from all outgoing edges ($\boldsymbol{E}^+$) is then assigned as the node's LCT. By beginning at the goal node (Z Shot) and working topologically backward, each node in the network is assigned its LCT after all its successors' LCTs have already been determined.

In this way, the STN from Anonymous (2017) can be extended to also include maximum times for activities, but unlike the lower bounds between activities derived from the *minimum physically possible* times, the upper bounds are derived from the *maximum operationally desirable* times (i.e., without sacrificing the overall operational goal for the day). The resulting network is similar to a Simple Temporal Network with Uncertainty (Vidal 1999), as it implicitly reflects "strong controllability" – though with the important difference that, in the present application, it is still entirely possible for activities' durations to violate the upper bounds, which violation would simply indicate that the operational goal can no longer be reached by the desired time given the structure of the network.

[2] Initial experiment start time or estimated time of resource availability can be used to determine ETEs for source nodes (i.e., no predecessors).

(More) Functional Distributed Reasoning

Through the construction of ETEs and LCTs, then, the present work aids in the goal of Anonymous (2017): that of participants' functional reasoning laid out by both Simon (1992) and Markus (2000), leading to more efficient coordination. While the present work extends the information incorporated and presented to the reasoning agents, it still continues the two means set out by Anonymous (2017):

1) Require as little information as possible from participants while still reliably modeling shot activities (e.g., only one common goal is required for calculation of all LCTs – though if a participant provides resource availability constraints, those can be easily incorporated), and

2) Provide consistently actionable information regarding alternative scenarios to Z SoS participants in order to aid them in their own planning, execution, adapting, and interfacing with other entities.

Including LCTs can provide SoS participants with actionable information to help coordinate work through reasoning that is more functional in several ways than the ETE-only STN constructed in (Anonymous 2017). First, it can provide at-a-glance information regarding slack in the experiment's schedule: when viewing a timeline, a participant can easily ascertain the window of time for each activity to begin, and the length of window for each activity's start time (Figure 1) relative to other windows can indicate that activity's proximity to the (anticipated) critical path. Second, it can provide an easy heads-up for those activities that will most impact the developing timeline of operations (and resultantly most impact the chances that the operational goal is achieved). If a participant perceives that the window for beginning an activity is uncomfortably small, they may adjust their behavior or the activity's scope, prepare in advance, garner additional resources, or even attempt to communicate/collaborate with others who may be impacted. Third, the bounded-window view compactly summarizes Simon's (1992) "alternative scenarios" (i.e., by showing a range of time over which each activity might happen, rather than a single prediction) – even showing the alternate ways an experiment is at risk of failing to achieve the operational goal. Through all of these means, this view increases understanding of the *behavioral* aspect of an experiment's schedule of activities for all participants (a la Rhodes & Ross 2010), and at once contributes to both the Upper Layer Description and Upper Layer Analysis of the SoS operations during a given day (a la Maier 2005).

One caveat in the present work's method of LCT calculation is that, since LCTs are derived from back-scheduling an operational goal through the minimum bounds of the net-work, the calculated LCTs are highly optimistic. If an activity is started at its latest cutoff time, the operational goal might still be able to be achieved, but only if all following activities' edges hold to their respective lower

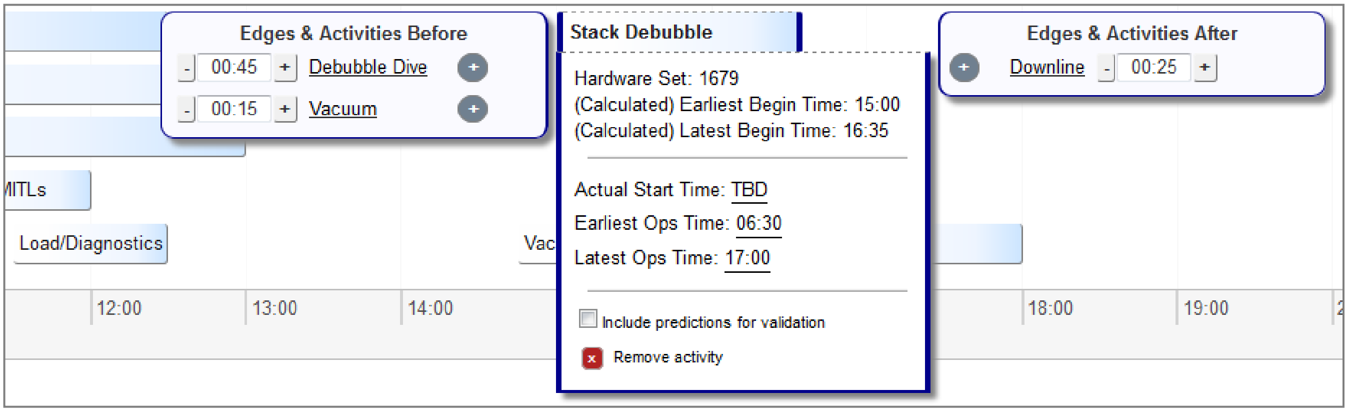

Figure 2. Meta-information available about an activity shown when a user selects that activity on the timeline. Edges and edge lengths are shown on each respective side of the info box, where they can be edited as needed (prompting rescheduling of the network). Calculated ETEs and LCTs as well as resource constraints (06:30 and 17:00) are shown in the info box.

bounds (each of which are individually optimistic). This optimism raises potential concerns for the perceptual aspect of complex systems interpretation (especially for unknown/unanticipated future participants in experiments, who may not realize just how optimistic the LCTs are). To help address this perceptual challenge, this high degree of built-in optimism to the LCTs should always accompany the LCTs through whatever means they are communicated, and the upper bounds of activities' windows should be clearly identified as challenging and undesirable locations to be occupying on the shot timeline – which is helped, for example, by including "cutoff" in the name.

Results

As discussed above in **Method**, the present work uses the minimum-bound STN constructed in (Anonymous 2017) to provide a template for a Z experiment. All activities are then scheduled to begin at their earliest times (thereby providing ETEs) on experiment day. The Z Shot activity is then defined as the operational goal, using a latest time determined by managers and experimenters. This goal node-time is then back-scheduled through the network in order to provide the upper bounds on activities' execution. The design of the database-driven web application created for use in storing, viewing, and modifying the plans and schedules of Z experiments is now discussed.

Creation of the Software Application

The scheduling of activities' ETEs and LCTs has been incorporated into an ASP.NET interactive web application, deemed PSYCHE (Planning with SYstematic CHronological Estimates). The resulting information system allows Z participants to capture and view information about activities before shot day, aiding them in making more informed decisions about their own work vis-à-vis the potential timeline developments of the experiment. Allowing the automated scheduling information to be viewed during planning stages is consistent with Smith's (2003) observation that in many applications, "planning…and scheduling…are not cleanly separable" and helps address the need Smith raises for "the design of more tightly integrated planning and scheduling processes" (2003). The vis.js framework (visjs.org) was used on the front-end to enable smooth, intuitive interactivity with the timeline, so that clicking on any activity on the timeline will provide more information about that activity, including valid operational hours of the activity as well as edges and lengths to other activities on the timeline. An example of this information can be seen in Figure 2.

In addition to aiding planners before an experiment is executed, PSYCHE aids experiment execution on shot day by including back-end interfaces with embedded facility diagnostics to provide real-time updates to the windows of time for which activities are scheduled. PSYCHE includes a scheduler that runs once each minute, updating the earliest times of all scheduled activities' windows to provide accurate up-to-the-minute information to shot participants on shot day. As real-time information comes in regarding when activities actually began (or as minutes pass by and activities do not begin), dependent activities' execution windows can be updated (i.e., their earliest time estimates shift to be later in time). These real-time updates do not affect the Latest Cutoff Times, since LCTs only depend on the predefined operational goal and the minimum edges defined in the temporal network. These minute-by-minute updates to an activity's ETE means that a scheduled activity's window "closes" as the day progresses, which is the way many participants already think about execution opportunities; therefore, the software's "closing window" depiction is in line with Markus et al.'s (2000) recommendation to match user intuition when possible.

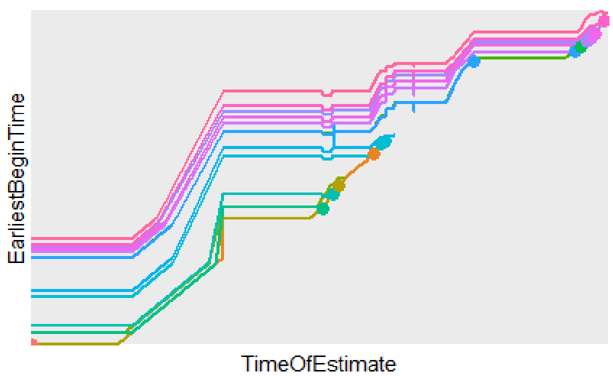

Figure 3. Traces of all activities' ETEs throughout a day; each line represents the trace of one activity's ETEs over the day (dot represents activity start). ETEs only increase as time increases.

Verification: Measuring Accuracy

One of the primary ways that PSYCHE's performance is being measured is by recording activities' ETEs minute-by-minute and comparing those estimates against activities' actual begin times. As discussed in **Method**, an ETE for an activity should always remain valid as an estimate of the *earliest* start time; if the estimate is adjusted, it should only be adjusted to be later in time (i.e., the left-hand side of an activity's execution window on the timeline should only move to the right), so that the original estimate remains valid as an earliest time (following Markus et al's (2000) recommendation to "improve offline behavior"). If any of the ETEs provided over a day for a given activity ends up being later than the actual start time of that activity, it is an inaccurate ETE. In addition, if any of an activity's ETEs estimate a time that is earlier than a previously provided ETE for that activity, then the previously provided ETE(s) should be considered inaccurate. By recording all of the minute-by-minute ETEs of an activity up to the actual begin time of that activity, it becomes possible to measure both of these conditions in order to check that the adjustments of estimates are behaving as desired (i.e., are only adjusted to estimate *later* start times as the experiment unfolds).

Motivated by the above observations, the calculated earliest times for a day's activities are recorded on a minute-by-minute basis throughout the day (since PSYCHE's real-time scheduler runs each minute, potentially updating the earliest times for each activity that day). The resulting records clearly show the behavior of estimates of all activities involved in a given experiment; an instance of the recorded minute-by-minute estimates for all activities in a shot can be seen in Figure 3. Each estimate can be seen to increase as a function of the time of the estimates, which is the expected and desired "accurate" behavior. The working definition for "accuracy" is defined for the present work as "the proportion of minute-based estimates of earliest begin time that are earlier than an activity's actual begin time."

In order to avoid surprise and encourage the check-in behavior discussed above in **Method**, it is necessary to maximize accuracy of the minimum bounds; however, it should be pointed out that one of the rather severe risks of achieving 100% accuracy by this definition is that *the minimum bounds between activities might be too small*. This would cause the LCTs of activities to be more optimistic than is appropriate (due to back-scheduling's use of minimum bounds). To address this risk, it may be desirable for some very small portion of estimates *to be inaccurate* – meaning an occasional activity's actual begin time is earlier than its minimum time estimate. Formalization of these latter concepts is ongoing, with the definition of "precision" of estimates being an area identified as a needed step in future work. In the meantime, the working definition of accuracy presently outlined serves as a practical measure of performance in serving the purpose of coordination.

Validation: Z SoS Participant Feedback

If Z participants are ostensibly those being served by the approach outlined in the present work, it stands to reason that they should be consulted on the perceived value of the work. Initial feedback along these lines has been obtained in several ways, albeit all anecdotal. First, casual conversations with installers and technicians have been conducted, inquiring as to which information would they rather be given: a "likely time" or even quantified probabilistic estimate, or a window of time during which an activity may occur. The results of these conversations fairly consistently reflect a desire for the "window" option.

Second, in response to direct questioning by a Z participant of "When will Activity X happen?", the response has been given in terms of a window of time (sometimes showing the PSYCHE timeline) and asked if that was satisfying. The answer was usually "Yes", though on occasion the reply was "Sure, but what time do you *think* it will happen?" (This latter response is not unexpected, since it reflects established cultural norms around gauging various individual perceptions in forming one's own opinion of likelihood, which is one of the asystematic behaviors that the present work is attempting to address.)

Finally, the timeline has been consulted and shown to decision makers on particularly complex shot days in order to communicate the slim margins of time associated with the operational goal that day. The decision makers were more informed by the visual timeline than they otherwise would have been, and were able to take action accordingly. Through continuing these types of interactions, it is hoped that feedback will continue to affirm the usefulness of the provided windows of time.

It is anticipated that as Z participants and decision makers interact with the visualization of PSYCHE, their perceptions and valuations of hypothetical and actual outcomes will change. This expectation is derived from previous applications of established psychology research to interactive visualizations, such as (Ricci et al. 2014). As discussed in that work, stakeholder's mental models of complex systems differ from the constructed models of systems used in engineering design. Interacting with visualizations of the constructed models (more specifically, being able to see the estimated results of different design choices), allows adjustment of both a stakeholder's mental model of a system and the constructed model of that system, leading to "better" decisions (defined by Ricci et al. as "trusted, truthful" decisions). It is further hypothesized in the present work that as Z participants observe and respond to the ETEs and LCTs provided to them (i.e., interact with the constructed model of an experiment), they will grow to trust the model and allow it to update their mental model as appropriate, leading to more consistent (and more confident) distributed functional reasoning

Further Work

With an initial prototype of PSYCHE complete, work can now transition to several fronts. First, the embedded sensors in the machine often provide false, conflicting, or irrelevant information. Real-time filtering and state estimation is needed to ensure that PSYCHE's estimates reflect the actual states of the machine. In tandem with the effort of creating such a filter/estimator, one of the next major improvements could be automated planning on subsets of activities to handle unscheduled events (e.g., rework) when the machine states indicate so.

Second, the activities chosen for the initial version of PSYCHE were chosen based on machine states which are already automatically diagnosed by embedded sensors. This set of states does not equal the set with which all Z participants are concerned, however. A more complete (and hopefully more broadly useful) state model is under construction, along with analysis of how embedded diagnostics could reliably indicate those states.

Finally, the methods of communication of the information captured and calculated by PSYCHE is an essential area ripe with opportunity. Live schedule updates could be communicated in multiple ways to various parties on the machine, increasing dissemination of progress and risks throughout a shot's execution. It is envisioned that this increased level of communication will further strengthen the interfaces between participants and implicitly encourage further cooperation.

Adding Probabilities and Entropy for Estimates

With a subset of the current states, and as more states are added, some of the temporal relationships between states will be able to be described reliably with probabilistic information. Because the current application does not incorporate any such information, it may need to be expanded in some way in order to provide as much useful information as possible when planning a Z experiment and increasing the chances of success in following a plan (even in the face of other activities/uncertainties that cannot be so characterized). One example of a potential improvement in this respect would be to implement a Probabilistic Simple Temporal Network with Uncertainty (PSTNU), which marries STNUs with probabilistic information so that a planner may incorporate as much information as possible to minimize risk (Santana et al. 2016). It is possible that a modified form of the PSTNU would allow more informed planning of Z experiments by experimenters interested in minimizing specific risks.

Importantly, however, this probabilistic information would only be intended to help planners before shot day: all of the challenges identified in **Method** remain for broad communication of such probabilistic information to the participant community, and it is not currently perceived that this added capability would contribute toward the present work's SoS-level goals of encouraging cooperation and leveraging interfaces for real-time execution.

An intriguing middle ground for broad communication of pseudo-probabilistic information might be an ordinal ranking of entropy for any ETE provided, so that if an activity were deemed fairly well known/described (e.g., automated fluid drains), successive activities' ETEs could be deemed "high quality", since less entropy is introduced into the temporal network from less-EKP-like activities. This measurement of "quality of estimate" could be useful to some participants, and those who are not helped by it could still safely ignore it, using only the ETE provided.

Adding System States

Another area of ongoing work is the expansion of the set of activities included in the planning and scheduling of Z shots. The currently included set comprises activities which are associated with already-existing embedded sensors. While the sensors are useful for the proof of concept of the present work, it is hypothesized that more useful states can be derived from stakeholder analysis and state machine studies, which are presently underway. The state machine(s) constructed will not only be able to inform the filtering of sensor input for more reliable real-time updates, but should also prove to be useful in any efforts toward automated planning on a subset of machine states.

Rolling Out Live Status Indicators

Once the initial version of PSYCHE has been operational for some time providing accurate estimates of the earliest times that activities can begin, those estimates (along with the latest times) can begin to be automatically communicated to Z participants. The aforementioned stakeholder analysis will likely aid in determining which types of participants need what information regarding schedule updates, as well as how often those updates are needed and the most effective methods for communication. Prior to the completed stakeholder analysis, various communication methods are already being considered, ranging from electronic kiosks with PSYCHE's view of the higher-level schedule for a shot, to LED matrix signs placed throughout workspaces, to automated email and PA announcements, among others. Since communication itself is one of the primary components of "the interfaces" between participants, the choices of what information to communicate, to whom to communicate that information, and the frequency and method of communication, have potentially drastic effects on the eventual success or failure of the present goals of leveraging interfaces and encouraging cooperation.

Conclusion

This work continues previous efforts to improve the interfaces of the Z System-of-Systems through distributed planning and automated scheduling of activities. Goals were defined for higher-level planning and scheduling activities to "leverage interfaces" and "encourage cooperation" by 1) requiring minimal information from each participant regarding their own planned activities, and 2) aiding in functional reasoning around the execution of activities for a given experiment. The present extension of the originally proposed STN is an STNU-like construct relating each activity with others, scheduling activities' Earliest Time Estimates (ETEs) and then using a single operational goal to provide the Latest Cutoff Times (LCTs) for all activities. The combination of ETE and LCT ultimately provide an execution window for each activity, which was incorporated into an interactive software tool, PSYCHE, intended to support both planning and execution. On-going areas of work were then discussed, including validation of estimates of earliest times, elicitation of participant feedback, and expanding the set of activities, sensors, and communications with which PSYCHE interfaces.

Acknowledgements

[Removed for double-blind reviewing].

References

Dechter, R., Meiri, I. and Pearl, J., 1991. Temporal constraint networks. Artificial intelligence, 49(1-3), pp.61-95.

Kahn, Arthur B. (1962), "Topological sorting of large networks", Communications of the ACM, 5 (11): 558–562.

Maier, M.W., 1996, July. Architecting principles for systems-of-systems. In INCOSE International Symposium (Vol. 6, No. 1, pp. 565-573).

Maier, M.W., 2005, October. Research challenges for systems-of-systems. In 2005 IEEE International Conference on Systems, Man and Cybernetics (Vol. 4, pp. 3149-3154). IEEE.

Rhodes, D.H. and Ross, A.M., 2010, April. Five aspects of engineering complex systems emerging constructs and methods. In Systems Conference, 2010 4th Annual IEEE (pp. 190-195). IEEE.

Ricci, N., Schaffner, M.A., Ross, A.M., Rhodes, D.H. and Fitzgerald, M.E., 2014. Exploring stakeholder value models via interactive visualization. Procedia Computer Science, 28, pp.294-303.

Santana, P., Vaquero, T., Toledo, C., Wang, A., Fang, C. and Williams, B., 2016. PARIS: a Polynomial-Time, Risk-Sensitive Scheduling Algorithm for Probabilistic Simple Temporal Networks with Uncertainty. Association for the Advancement of Artificial Intelligence.

Anonymous, 2017. Details omitted for double-blind reviewing.

Simon, H.A., 1996. The sciences of the artificial. Cambridge, Mass.: MIT press.

Smith, Stephen F. (2003). Multidisciplinary Scheduling: Theory and Applications. In 1st International Conference, MISTA '03 Nottingham, UK (Vol. 13, No. 15, pp. 3-17).

Vidal, T., 1999. Handling contingency in temporal constraint networks: from consistency to controllabilities. Journal of Experimental & Theoretical Artificial Intelligence, 11(1), pp.23-45.