SAND2016-11860C

# A Simple Temporal Network for Coordination of a Collaborative System-of-Systems in Research Operations with Large Exogenous Uncertainties

**Michael A. Schaffner**

Sandia National Laboratories
Albuquerque, NM
maschaf@sandia.gov

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

**Abstract**

The Z Machine is the world's largest pulsed power machine, 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 with independent functions and management, forming a Collaborative System-of-Systems. This structure, combined with the research-oriented nature of experiment preparations, creates significant challenges in planning and coordinating required activities leading up to a given experiment. The present work demonstrates an approach to scheduling planned activities and presenting information relevant on "shot day" to aid as an enabling interface between workers among these different groups, using minimal information to begin forming a Simple Temporal Network with Uncertainty (STNU). First, a simplified model of an experiment comprising common shot activities is defined, with the minimum physically possible times (i.e., lower bounds) between those activities described. The difficulty of determining maximum possible times (i.e., upper bounds) between these activities is discussed, concluding with the use of a single operational goal to back-schedule all latest times of when activities must begin to achieve that goal (so that unlike the lower bounds which are minimum physically possible intervals, the latest times reflect maximum operationally desirable times). The resulting STNU-like product's use in real-time operations is then presented, and further work discussed.

## 1 Introduction

The Z Machine (hereafter "Z") is the world's largest pulsed power machine, 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. Regular operation of Z on a daily basis requires specialists from these fields as well as technicians and installers performing regular machine maintenance and configuration, which involves activities such as operating heavy machinery, refurbishing equipment, performing routine mechanical and electrical work, and even underwater diving, among others.

The activities, specialties, and organizations involved in Z experiments and operations have evolved over time, posing significant challenges to coordination of activities using static plans and schedules. While much of the funding for the experiments and operations of the machine comes from a single organization, many activities and capability enhancements are funded at least in part through alternate sources. In addition, many of the supporting staff for diagnostics, targets, and subsystems have independent management and volunteer-like participation with Z experiment preparation and execution. These traits, specifically varying levels of "operational independence" and "managerial independence" of components, place Z on the spectrum of a Collaborative System-of-Systems (SoS) (Maier, 1998). This type of operation has no recognized central authority to provide top-down guidance on organization and execution of work, and often there exist no centrally or commonly defined roles and responsibilities. While individual sections and agents may generate their own activities and associated (implicit or explicit) plans and schedules for those activities, such plans and schedules may be communicated in an ad-hoc manner or simply adapted in-situ pursuant to the present experiment's perceived progress. Such behaviors (i.e., ad-hoc communication and in-situ adaptation) significantly challenge efforts in higher-level planning and scheduling for experiments to aid in coordination across groups; even if fully informed (which is rarely the case), static plans and schedules – even those created very close to "shot day" – can quickly become obsolete, causing wide-varying interpretations of any schedule updates or future schedules.

## 1.1 Motivation: Enabling Coordination

Despite the challenges posed to planning and scheduling, many stakeholders and participants in the Z SoS consistently express a desire for a higher-level understanding of the system's anticipated and actual temporal behavior for a given experiment. To put it in the simplest terms, the two main questions that sum up most concerns are a form of, "How do we think we're going to do?" (before shot day) and "How are we doing?" (during shot day). One of the primary challenges to answering these questions with a higher-level plan/schedule is that individual agents choose when to execute activities based on their own real-time perception – stemming from a variety of self-determined indicators – of how other activities and events are proceeding. The interfaces between participants on a given shot are sometimes known in advance but, as mentioned above, are often of an ad-hoc nature. Eliminating this behavior is not possible, nor is it desirable, since in fact this ability to adapt is widely recognized as essential to the success of Z experiments due to the research-oriented (and therefore often unknown) nature of the work.

This research-intensive environment poses another major challenge to higher-level planning and scheduling on Z, however. Many shot activities represent active areas of research, including the primary machine's regular performance (e.g., delivery of electrical current), regular diagnostics (e.g., x-ray measurement), and experimental subsystems and diagnostics (e.g., plasma cleaning, CMOS cameras). Activities are often planned which have no clear upper bound of time associated with them, whether because they involve completely novel apparatus or procedures, or because the effects and timing of the activity have not been well-characterized by statistical methods and measures (or *cannot be* due to epistemic uncertainties). This inability to constrain operations activities' timings to well-characterized, limited-duration events provides the second significant challenge to higher-level planning and scheduling of activities for a given experiment.

Even with these two major challenges present, the need remains for a consistently defined, unambiguous presentation of an experiment's events before shot day (e.g., for enabling planning and coordination ahead of time) and during shot day (e.g., for enabling in-situ adaption). In keeping with Maier's architectural principles for an SoS, this endeavor is a form of "leveraging interfaces" of and "ensuring cooperation" by all parties involved in the Cooperative SoS (1998). But when designing a Z experiment, many activities can be planned to happen simultaneously, and uncharacterized exogenous uncertainties surround many of the activities' timescales, so it is difficult to estimate in advance the impacts of one or more additional activities or the uncertainty that exists when planning ahead for and adapting during an operational day. For this reason, when designing an experiment, it is desirable to understand the *behavioral aspect* by modeling "the emergent behaviors resulting from these complex interconnections in order to understand how the system will perform" (Rhodes and Ross 2010). (For the present work, the scope of behavior is limited to temporal behavior.)

Equally important to enabling coordination among independent participants, however, is understanding the *perceptual aspect*, which

> …relates to how the system is interpreted through the perspective of system stakeholders. This aspect considers individual stakeholder preferences, and how preferences vary across stakeholders. It also considers the changes in preferences as a response to context shifts over time as the stakeholders interact with the system in its environment. This aspect relates to cognitive limitations, biases, and preferences of the stakeholders (2010).

This latter aspect of the problem implies that 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*.

## 1.2 Pitfalls of Naïve Prediction

A common question that most Z experiment participants have asked at some time or another is, "When is Activity X going to happen?" And indeed, a naïve goal of constructing a schedule may be to try to answer this question in the context of the Z SoS, even with the challenges presented above. However, since most participants agree that static predictions like this question cannot be consistently accurate, many ask instead a question which looks less naïve because it invokes probabilistic measures: "When is Activity X likely to happen?" Due to the unique characteristics of Z as a research-intensive Cooperative SoS, however, this question is also naïve. First, there exist little statistical data on which to base probabilistic estimates for most of the activities, and requiring data (or estimates) from all parties involved neither encourages cooperation nor ensures verified/validated data. Second, in a research-intensive environment like Z, where many epistemic sources of uncertainty are present, statistics prove to be very poor measures of temporal behavior. Even when characterized, the meaning of long tails, extreme skewness, and multiple modes of distributions vis-à-vis planning/scheduling are virtually impossible to communicate individually, much less in aggregate form, to all participants in the SoS. Finally, though, and most critically, keeping in mind the *behavioral* and *perceptual* aspects of an

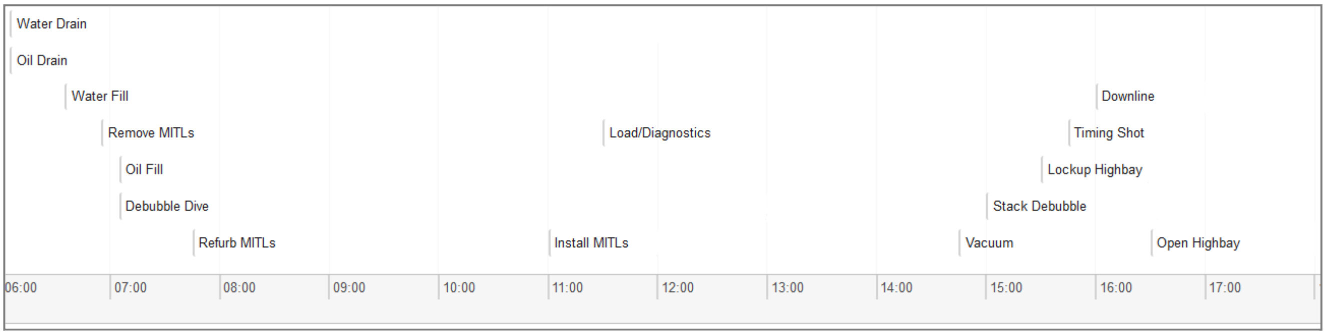


*Figure 1. First step of the present method: scheduling of a reduced Z Simple Temporal Network based on earliest times. (Vertical spacing/proximity is only a function of the non-overlapping layout algorithm used.)*

experiment, the answer to this question of an activity's "likely time" 1) will not stay constant throughout an experiment, 2) may be different for every participant in the SoS (due to their perceptions when quantifying "likely"), and 3) will be of varying degrees of usefulness to every participant (due to cognitive limitations, biases, and preferences). Perhaps most importantly, providing "likely" times may not even encourage behavior that aids in the overall system's functioning, since there is no (and cannot be a) centrally defined "correct" response to probabilistic information in an environment with independent management and operational behaviors.

## 2 Method

Simon (1992) acknowledges the dangers inherent in attempting to answer predictive questions like those above when he writes, "Because of the possible destabilizing effects of taking inaccurate predictive data too seriously, it is sometimes advantageous to omit prediction entirely." Predictions can help participants in some environments, but the goal of the present work – in keeping with Maier's recommendations – is to provide information that encourages participants to cooperate with the wider system in planning, executing, and adapting their own work. In pursuing this type of goal, Simon writes, "Numbers are not the name of this game but rather representational structures that permit functional reasoning, however qualitative it may be…The heart of the data problem for design is not forecasting but constructing alternative scenarios for the future…" (1992). Functional reasoning is the goal Simon outlines: in the present application, the function being overall SoS coordination and interfacing of constituent members. The present work, therefore, pursues two means of achieving that goal: 1) *require as little information as possible from participants* while still reliably modeling shot activities (e.g., do not require statistical distributions generated from large empirical datasets), 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.

A Simple Temporal Network with Uncertainty (STNU) seems a natural fit for these two goals, due to its relatively lightweight data requirements, its ability to aid in functional reasoning regarding potential timeline developments (e.g. windows of execution, controllability of a schedule), and its use of upper and lower bounds to capture uncertainty in durations between activities (Vidal 1999). The minimum bounds between activities on Z can in most cases be quite easily ascertained, as participants are usually quite able to provide an optimistic (and often even realistic) estimate of how fast work can be done, even work which has never been performed before. As previously discussed, however, upper bounds on activity durations are often much more difficult to elicit and implement in practice. Therefore the present work begins based on Dechter's (1997) work by simply using the minimum possible times between activities to construct a Simple Temporal Network (STN), defined as a tuple {**V**, **E**}, where

1) **V** : set of nodes representing activities (e.g., "Fill Water")
2) **E** : set of edges of the form $l < dest - src$ , where
   $dest,src \in \mathbf{V}$
   $l \in \mathbb{R}_{>0}$
3) In the present application, no cycles exist.

Once this network is created, all activities can be scheduled at their earliest possible times for a given experiment (see Figure 1). 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 then be defined (e.g., "Fire the machine by 5pm today"), which can be back-scheduled through the network to provide the upper temporal bounds for all preceding activities (see Figure 2). In this way an STNU-like network can be constructed, but

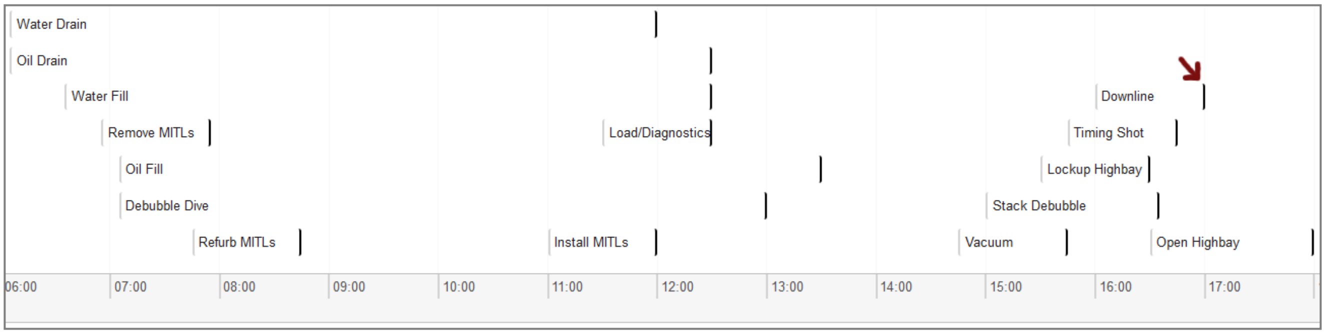


*Figure 2. Physically possible earliest start times are shown by gray lines on the left side of each activity (same as Figure 1). Bold arrow indicates the operational goal for the day used in back-scheduling, resulting in the latest "operationally possible" start times, shown by black lines on the right side of each activity. (Vertical proximity/spacing is only a function of the non-overlapping layout algorithm used.)*

with the lower bounds between activities representing the minimum physically possible times, and the upper bounds determined by latest times which are the *maximum operationally possible* times (i.e., without sacrificing the overall operational goal for the day). While this network is not a proper STNU, it does implicitly reflect "strong controllability" (i.e., only the upper bounds of the contingent constraints are known in advance, since they are calculated based on the operational goal set for the experiment's execution).

### 2.1 Result: Distributed Functional Reasoning

The STNU-like network that results from this approach can provide SoS participants with actionable information to help coordinate work through functional reasoning in several ways. 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 2) derives from that activity's proximity to the (estimated) critical path. 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). It also compactly summarizes Simon's "alternative scenarios" (i.e., by showing a range of time over which each activity might happen, rather than a single prediction). Through these means, this view increases understanding of the *behavioral* aspect of an experiment's schedule of activities for all participants.

Second, the resulting network provides an earliest time for participants to "check-in" on shot day for any given activity of concern. Though the earliest time of an activity (i.e., window's begin time) may change on shot day, it should only get pushed back in time, meaning that the act of "checking in" will be informative to participants either way (i.e., either the activity will be ready for them to participate in, or the participant will get an update of when next to check-in). This assurance of useful information encourages behavior similar to complex sociotechnical systems like buses and airlines, where a minimum time is given to coordinate many participants in "checking in", but the estimated time of the event might be modified (usually to be later in time, almost never earlier) from the one originally given in order to accommodate large exogenous uncertainties. This result directly addresses both the *behavioral* and *perceptual* aspects of communicating higher-level scheduling information.

Finally, the STNU-like network can provide important information about the latest time to begin an activity. In the present work's application, where the latest times are derived from back-scheduling an operational goal through the minimum bounds of the network, the calculated latest time is highly optimistic. The reason for it being highly optimistic is that if an activity is started at its latest time, the operational goal can still be achieved, but only if all following activities' edges hold to their respective lower bounds (each of which are individually optimistic). Because of this high degree of optimism, participants should be looking to begin activity execution well before the end of the window (i.e., at least "check-in" for an activity at the start of its window), directly addressing the *perceptual* aspect of the problem of prediction.

## 3 Application

As discussed above in Method, the present work begins by modeling shot activities and minimum times between activities. This information is then used to construct an STN, which can schedule all activities to begin at their earliest times. An operational goal is then defined for firing the shot, allowing the latest time for the shot to be back-scheduled through the network in order to provide the upper bounds on activities' execution. The present work concludes with a database-driven web application being creat-

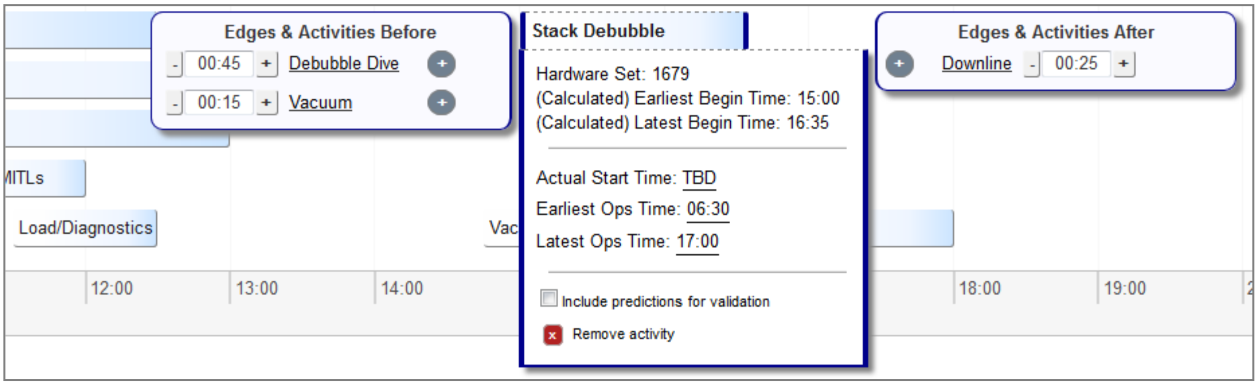


*Figure 3. PSYCHE's interface for viewing an activity's meta-information.*

ed for use in storing, viewing, and modifying the plans and schedules of Z shots.

### 3.1 Creating and Scheduling the STN

A reduced model of a shot was created for the initial proof of concept, comprising 15 activities across 5 independent groups. The activities chosen were based on operations diagnostics that automatically update machine states based on electromechanical and electronic triggers throughout the Z machine. The minimum-time edges could then be derived from the electronic records of the changing states of the machine. Three example activities and their minimum-time edges are listed below in Figure 4. The STN created can be used to schedule all activities at their earliest begin times, shown in Figure 1.

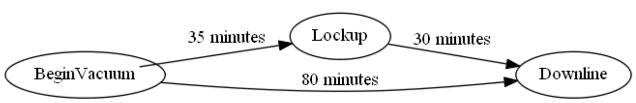


*Figure 4. Three activities and minimal temporal relationships.*

### 3.2 Calculating Latest Times

Once all shot activities are scheduled at their earliest beginning times, an operational goal for firing the shot is established. For example, based on Downline's earliest time in Figure 1, an operational goal can be established of "go Downline by 5pm." This provides a latest time for the Downline activity, which can then can be back-scheduled throughout the network (using the already-defined minimum times between activities) to establish latest times for each activity (see Algorithm 1).

Once latest times have been established for all activities, the "windows" for each activity are complete. The result can be seen in Figure 2, where the left side of each window is scheduled at the earliest <u>physically</u> possible time that the activity can begin, and the right side of each window reflects the latest <u>operationally</u> possible time that the activity can begin (i.e., if the operational goal has any chance of holding).

### 3.3 Creating the Software Application

A database-driven web application was created with SQL, C#, and HTML/CSS/Javascript to allow the planning of activities for an experiment while visualizing the schedule for the planned activities in the manner described above. The vis.js framework (visjs.org) was used 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 3. The application, known as PSYCHE (Planning with SYstematic CHronological Estimates), 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.

In addition to addressing "How do we think we will do?" before shot day, PSYCHE also tackles the question

**<u>Algorithm 1: CALCULATE LATEST TIMES</u>**

**Input:** Graph G = {V, E}
**Output:** V with each *v.latestTime* assigned
**Function** CalcLatestTimes(G)
  V ← TopologicalSort(V)
  reverse(V)
  **for** *v* in V **do**
    *v.latestTime* ←
      min [ *e.dest.latestTime* – *e.l* ]
      ∀ *e* in edgesFrom(*v*)
  return V

"How are we doing?" by including back-end interfaces with the aforementioned embedded machine 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 execution times shift to be later in time). These real-time updates should not affect the latest times that activities can be scheduled, since those only depend on the predefined operational goal. These minute-by-minute updates to the earliest possible times of activities means that a scheduled activity's window "closes" as the day progresses, which matches established intuition of participants involved with a shot.

## 4 Results and Continuing Work

With an initial operational version of PSYCHE recently rolled out, work has now transitioned to several fronts. First, measurement of the success of the present work can begin to be conducted. Clearly, measuring the success of any endeavor to "leverage interfaces" and "encourage cooperation" is challenging due to the lack of consistent interpretation of these terms. Nevertheless, several indicators are presently proposed by which to gauge the progress and level of success of the current work.

Second, the embedded sensors in the machine occasionally 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 will be automated planning on subsets of activities to handle unscheduled events (e.g., rework) when the machine states indicate so.

Third, 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.

### 4.1 Defining Accuracy

One of the primary ways that PSYCHE is being measured is by recording activities' calculated earliest begin times and comparing those estimates against activities' actual begin times. As discussed in **2.1**, a calculated earliest time for an activity should always remain valid; if it 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 right), so that the original estimate remains valid as an earliest time. If any estimate of an earliest time for an activity ends up being later than the actual time of that activity, this would indicate an inaccurate estimate. By recording all of the calculated earliest begin times of an activity up to the actual begin time of that activity, it becomes possible to check that the adjustments of estimates are behaving as desired.

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 example of the recorded minute-by-minute estimates for all activities in a shot can be seen in Figure 5. 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 in **2.1**, it is necessary to maximize accuracy; however, it should be pointed out that one of the

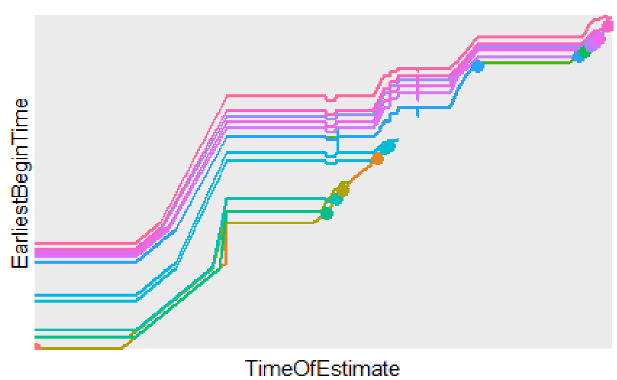


*Figure 5. Traces of all activities' estimated begin times throughout a day; each line represents the trace of one activity's estimates over the day. Estimates only increase as time increases.*

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 latest time estimates of activities to be more optimistic than is appropriate (see Algorithm 1'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 minimum time estimate is later than an activity's actual begin time. 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.

## 4.2 Eliciting 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 cultural momentum 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 estimated earliest times 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.

## 4.3 Adding System States and Probabilities

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.

As more states are added, some of the temporal relationships between states will likely 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 optimizing 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 participants interested in minimizing risk.

## 4.4 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 stakeholder analysis proposed in **4.3** 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 how and how often to communicate that information, have potentially drastic effects on the eventual success or failure of the present goals of leveraging interfaces and encouraging cooperation.

## 5 Conclusion

This work began by classifying Z machine experiments as a System-of-Systems with varying levels of managerial and operational independence. 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 method chosen to achieve these goals was an STNU-like construct that relates each activity with others, schedules activities' earliest possible times, and then uses a single operational goal to provide the latest possible start times for each activity, providing an execution window that reflects the earliest physically possible times and the latest operationally desirable times. A simplified model of a Z experiment was shown, and PSYCHE, the software application that implements the STNU-like results for the simplified Z experiment, was then presented. 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.

## Acknowledgments

The author is grateful to Sandia National Laboratories for providing the funding for this research. The author is additionally grateful to Ryan Kamm for feedback and encouragement, and to Trent Yocom and Rafael Aragon for their knowledge and patience.

## References

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

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

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.

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

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