ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THMPL01

A SIMPLE TEMPORAL NETWORK FOR COORDINATION OF EMERGENT KNOWLEDGE PROCESSES IN A COLLABORATIVE SYSTEM-OF-SYSTEMS

[1] M.W. Maier, "Architecting principles for systems-of-systems," Systems Engineering, vol 1, no. 4, pp. 267-284, 1998.

M. A. Schaffner†, Sandia National Laboratories, Albuquerque, NM, USA

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 Emergent Knowledge Processes central to preparation and execution, 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 on "shot day" to aid in coordinating workers among these different groups, using minimal information about activities' temporal relationships to form a Simple Temporal Network (STN). Historical data is mined, allowing a "standard" STN to be created for common activities, with the lower bounds between those activities defined. Activities are then scheduled at their earliest possible times to provide participants a time to "check-in" when interested.

INTRODUCTION

"Linearity is an artificial way of viewing the world. Real life isn't a series of interconnected events occurring one after another like beads strung on a necklace."

– Ian Malcom, in Jurassic Park

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 electromechanical controls. 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.

Challenges to Coordination

The activities, specialties, and organizations involved in Z experiments and operations have evolved over time, posing significant challenges to coordination of daily activities using static and deterministic 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 and organizations, leading to varied and dynamic relationships between participating personnel and systems. 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, especially varying levels of "operational independence" and "managerial independence" of constituents, place Z on the spectrum of a Collaborative System-of-Systems (SoS) [1]. 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 perceived progress of a given experiment throughout a day. 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; static plans and schedules – even if fully informed (which is rarely the case) and even if created very close to "shot day" – can quickly become obsolete, causing wide-varying interpretations and even distrust of any schedule updates or future experiments' schedules.

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 emergent) nature of much of the work. Such work is typical of Emergent Knowledge Processes (EKPs), which "involve intellectual activities, expert knowledge, and diverse people in unstructured and unpredictable combinations" [2].

† maschaf@sandia.gov

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THMPL01

This emergent knowledge 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 insufficient data and/or 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.

Despite these two major challenges 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). These two questions reflect a common need for a consistently defined, unambiguous presentation of an experiment's events *before* shot day (which would better enable planning and coordination ahead of time, as well as provide an indicator for likelihood of success) and *during* shot day (to better enable adaption and collaboration, as well as increase the likelihood of success). In keeping with Maier's architectural principles for an SoS, endeavoring to answer these questions is a form of endeavoring to "leverage interfaces" of and "ensure cooperation" by all parties involved in the Cooperative SoS [1]. When designing a Z experiment, many activities can be planned to happen simultaneously, and uncharacterized (i.e., epistemic) uncertainties surround many of the activities' timescales, so it is difficult to accurately 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" [3]. (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. [3]

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*.

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 deterministic 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 Cooperative SoS centered around Emergent Knowledge Processes, 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 with many EKPs, where epistemic sources of uncertainty have large effects, aleatoric distributions (i.e., statistics) often prove to be unhelpful descriptors of temporal behavior due to the overwhelming effects of the uncharacterized portions of uncertainty. 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. Keeping in mind the *behavioral* and *perceptual* aspects of an experiment: the answer to this question of an activity's "likely time" will not stay constant throughout an experiment, the answer may be different for every participant in the SoS due to their perceptions when quantifying "likely" [4], and the answer will be of varying degrees of usefulness to every participant due to cognitive limitations, biases, and preferences. Perhaps most importantly, <u>providing probabilistic times does not encourage behavior that aids in real-time coordination [5]</u>, since it is always preferable for resources to be available *ahead of time*. Finally, there is no (and cannot be a) centrally defined "correct" response to probabilistic information in an environment with independent management and operational behaviors.

METHOD

Herbert Simon points out the danger inherent in attempting to answer predictive questions like those above when he writes, "Because of the possible destabilizing

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THMPL01

effects of taking inaccurate predictive data too seriously, it is sometimes advantageous to omit prediction entirely" [6]. Predictions can help participants in some environments, but the goal of the present work – in keeping with recommendations of [1] – 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, "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…" [6]. "Functional reasoning" is the goal outlined: 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 sufficiently large empirical datasets), and 2) *provide consistently actionable information regarding alternative scenarios* to Z SoS participants in order to aid them in their own plans, execution, adaptation, and interfacing with other entities.

A Simple Temporal Network [7] seems a natural fit for these two goals, due to its relatively lightweight data requirements and its ability to aid in functional reasoning regarding potential timeline developments. The minimum bounds between an activity and its successors in a Z experiment can in most cases be quite easily ascertained, as participants are usually quite able to provide an optimistic (and often even realistic) estimate of the fastest time in which an activity can be completed, even activities which have never been performed before. Therefore the present work begins based on [7] by using these minimum possible times between activities to construct a *directed constraint graph* with universal (infinite) upper bounds on all intervals, leaving a constraint graph with only minimum bounds. This constraint graph can then be converted to a *distance graph*, which is a directed edge-weighted graph **G** defined as a tuple **G** := {**V**, **E**}:

V: set of nodes, each representing the start of activities (e.g., "Begin Water Fill")
E: set of edges representing the minimum minutes between nodes, of form $dest_{start} - src_{start} \geq a$, where
$dest, src \in \mathbf{V}$
$_{start}$ = start time of activity
$a \in \mathbb{R} > 0$
No cycles exist.

A simplified example of a distance graph comprising 3 vertices is shown in Fig. 1. Once such a graph (a Simple Temporal Network, or STN) is created, it can be used to schedule activities relative to one another by simple addition of the temporal constraints between nodes.

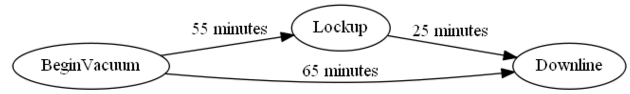

Figure 1: Simple Temporal Network (STN) comprising 3 nodes and 3 edges.

Creating and Scheduling the STN

A reduced model of a shot was created for the initial proof of concept, comprising 15 activities across 6 independent groups. The activities chosen were based on operations diagnostics that automatically record machine states based on electromechanical and electronic triggers throughout the Z machine. As an example, a plot of hundreds of past times between two activities' start times – vacuum and a downline shot – is shown in Fig. 2.

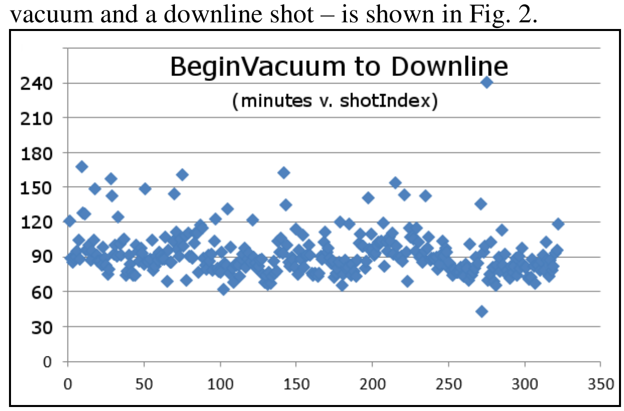

Figure 2: Time in minutes between starting vacuum and a downline shot, for a few hundred shots (indexed 0-330).

The minimum-time edges can then be derived from these electronic records of the states of the machine; in Fig. 2's case, the minimum time could be estimated to be just over 60 minutes (to derive reliable minimum times from such records, some judgment is necessary to adjust for outliers). This estimate can then form the edge in the STN between these two activities, and analysis and construction of the remaining activities and edges follows the same pattern. The complete STN created for all 15 activities and their relationships can then be used to schedule all activities at their earliest begin times, shown in Fig. 3.

Result: Distributed Functional Reasoning

The STN that results from this approach can provide SoS participants with actionable information to help coordinate work through functional reasoning in several ways. First, it helps compactly summarize the "alternative scenarios" recommended in [6] by simply showing a lower-bounded range of time over which each activity might happen, rather than a single prediction. This type of summary 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. An earliest time estimate provided earlier in time will not be invalidated by later modifications to the activity's earliest time estimate, since by defi-

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THMPL01

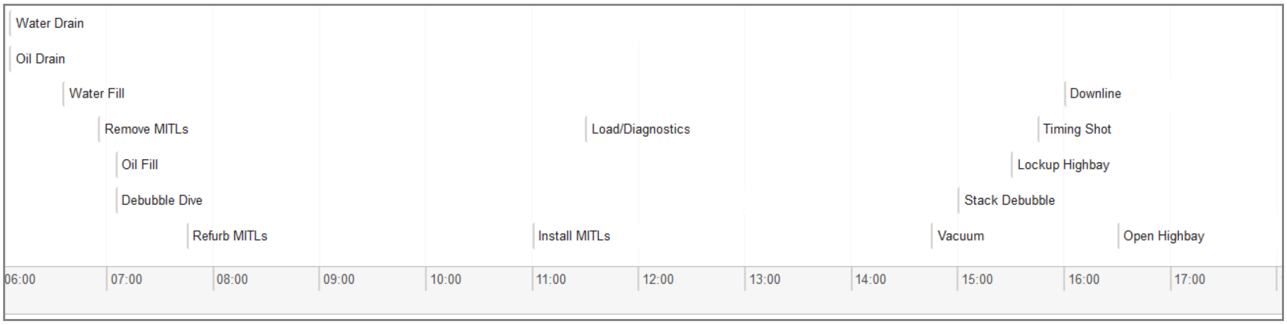

Figure 3: Scheduling of a reduced Z Simple Temporal Network based on earliest times activities could start given an operations start time of 6am. (Vertical spacing/proximity is only a function of the non-overlapping layout algorithm).

nition the estimate should only get pushed *later* 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 at the estimated time, or the participant can 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 therefore helps directly address both the *behavioral* and *perceptual* aspects of communicating higher-level scheduling information.

FURTHER WORK

The present work can be expanded on in several ways presently proposed. First, the STN could be greatly improved with the incorporation of upper bounds on activities' temporal relationships, to help provide not only earliest estimated start times but also *latest* estimated start times of activities. Some regular machine activities do have reliable upper limits on how long they might take, but even one activity without a definite upper bound (of which activities there are many in EKPs) would prevent any estimate of latest start times for all downstream activities in the STN. Probabilistic information may help address this problem in some fashion but is not viewed as an ideal solution given the discussions above in *Pitfalls of Naïve Prediction*. In addition, upper bounds intermittently or inconsistently incorporated into the STN could confuse more than help participants, since the information guarantees discussed above would no longer hold true. Further work could potentially address this opportunity for improvement in finding a suitable method to incorporate upper bounds.

Another area for further work is the STN's development into a participant-facing software tool that would serve as a display of the STN for a given experiment. Activities and their minimum estimates could be added/removed in advance by participants or an administrator, allowing a more well-informed system-level view of an experiment in advance of execution. In addition, if the software tool were then connected to the machine state sensors on which the activities are based, then the STN could be automatically rescheduled as each activity begins (or doesn't), serving as a real-time display that participants could reference throughout an experiment to better help coordination (and to encourage "checking in" on any connected device) as an experiment progresses.

CONCLUSION

This work began by classifying Z machine experiments as a System-of-Systems with varying levels of managerial and operational independence executing activities that include many Emergent Knowledge Processes. 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 a Simple Temporal Network that temporally relates each activity with its predecessors and successors, allowing activities to be scheduled at their earliest possible start times. A simplified model of a Z experiment was created, and an example schedule was shown. Further work was then discussed, including the challenge of incorporating upper bounds into the network and the creation of a software tool to help administer and communicate the results of the STN's automatic scheduling/rescheduling as an experiment progresses.

ACKNOWLEDGEMENT

The author is grateful to Ryan Kamm and the workers at Z for feedback and encouragement.

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International Inc. for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

REFERENCES

[1] M.W. Maier, "Architecting principles for systems-of-systems," Systems Engineering, vol 1, no. 4, pp.267-284, 1998.

ISBN: 978-3-95450-193-9 doi:10.18429/JACoW-ICALEPCS2017-THMPL01

[2] M. L. Markus *et al.*, “A design theory for systems that support emergent knowledge processes”, MIS Quarterly, pp.179-212, Sep. 2002.

[3] D.H. Rhodes and A.M. Ross, “Five aspects of engineering complex sysems emerging constructs and methods”, in IEEE Systems Conference 2010, pp. 190-195.

[4] D. Wark, “The Definition of Some Estimative Expressions”, Studies in Intelligence, vol. 8, pp. 64-80, Fall 1964.

[5] S.L. Savage & H.M. Markowitz, “Introduction,” in The flaw of averaes: Why we underestimate risk in the face of uncertaity. Hoboken, NJ, USA: Wiley, 2012, pp. 1-2.

[6] H.A. Simon, The sciences of the artificial. Cambridge, MA, USA: MIT press, 1996.

[7] R. Dechter *et al.*, “Temporal constraint networks,” Artificial intelligence, vol. 49, no. 1-3, pp. 61-95, 1991.