# TIPS – A system for contextual prioritization of tactical messages

Dr. Robert E. Marmelstein, Dr. Mary G. Devito
Department of Computer Science
200 Prospect Street
East Stroudsburg University,
East Stroudsburg, PA 18301

Dr. Mark H. Linderman
AFRL/RISE
525 Brooks Rd
Rome, NY 13441

Dr. Norman Coleman –
AMSRD-AAR-WSF-N
Intelligent Systems Branch,
Network Lethality & Int. Systems Div.,
Picatinny Arsenal, NJ 07806

# TIPS – A system for contextual prioritization of tactical messages

Dr. Robert E. Marmelstein, Dr. Mary G. Devito – East Stroudsburg University, East Stroudsburg, PA
Dr. Mark H. Linderman – Information Directorate, Air Force Research Laboratory, Rome, NY.
Dr. Norman Coleman – Army Research and Development Center, Picatinny Arsenal, NJ.

**Abstract:**
The automatic understanding of military messages has long been a technological goal of Command and Control (C2). Many past approaches utilized Natural Language Processing (NLP) techniques to solve this problem. However, emerging database schemas for coalition C2, such as the Joint Command, Control and Consultation Information Exchange Data Model (JC3IEDM), make other approaches feasible. The richness of the JC3IEDM makes it possible to extract context for Battlespace objects represented in the database. The Tactical Information Prioritization System (TIPS) leverages the JC3IEDM schema in order to automatically evaluate and prioritize tactical reports for transmission to a given unit. Priorities are assigned based on discovered linkage relationships between the report and the unit of interest in a number of different categories. Once found, this evidence is injected into a specialized Bayesian network to generate the relative report priority. In this paper, we describe the conceptual framework behind TIPS, including key queries to determine message relevance to a given unit. We also present results showing scalability of the TIPS for a variety of scenarios.

## 1. Introduction

The problem of getting the right information to the right person at the right time is as old as warfare itself. Despite the development of elaborate C2 systems and strategies, the solution remains elusive. Part of the problem is information overload; that is, C2 systems generate increasing volumes of information, which (in turn) needs to be evaluated, categorized, and given a disposition. At the end of the day, there are just not enough C2 personnel to make timely decisions about who should get what.

Given this, it is highly desirable to have automated systems that can evaluate and prioritize information. Significant effort has gone into NLP systems that can understand text messages. Perhaps the most prominent of these initiatives are the series of Message Understanding Conferences (MUC) sponsored by Defense Advanced Projects Research Agency (DARPA) from 1987-1997 [6]. While significant progress was made in message understanding [2], this is only part of the problem. What is also needed is a mechanism for assessing the context of the message relative to other Battlespace entities, such as military units. In short, messages for a given unit should be prioritized based their content relative to the unit's context.

On first glance, this may appear an impossible task for anything other than a person to perform—and not just anyone. Only a small subset of C2 personnel might be able to perform such a task well—those with a comprehensive, up to the second, understanding of what is going on in the Battlespace. Even more, to make the accurate decisions, personnel also need to understand the intricate relationships between different Battlespace entities: who is doing what and how is it related? Few people, other than perhaps top level commanders, have such knowledge—and they tend to be rather busy. Other factors, like stress and sleep deprivation, also serve to reduce the efficiency of C2 personnel [3]. For these reasons, the degree of information overload in operations like Kosovo has challenged our ability to achieve information superiority [4]. The frequent result is that either too little or too much information gets to the intended recipient.

Beyond message transmission, prioritization is also important for presentation of information—making sure critical information is brought to the user's attention. This is especially important when the amount of presentation space is limited. An example of this limitation is the squad leader who relies on a mobile, hand-held device for Situational Awareness (SA). Only so much information can fit on the display—as a result, the most important information must stay visible.

## 2. Approach

In this paper, we introduce the Tactical Information Prioritization System as a mechanism for prioritizing messages for transmission to military units within a Battlespace. TIPS accomplishes this task by mining linkages from a sufficiently structured database schema. Modern C2 database schemas, such as the JC3IEDM[1], provide a ready source for this purpose. By extracting context from a JC3IEDM instance, we avoid much of the complexity involved with NLP-type approaches. TIPS is made up of three essential components: a C2 database schema, a set of queries to discover contextual relationships (linkages) between the Reported Data Item (RDI) and the Unit of Interest (UOI), and a Bayesian Network (BN) to compute a priority based on the number and type of discovered linkages. We now look at each of these components in turn.

### JC3IEDM

Perhaps the most difficult and fundamental part of the message prioritization problem is how to capture Battlespace context. While there are many C2 database schemas that capture aspects of context, until recently none did so in a comprehensive way. That changed with the introduction of the JC3IEDM. The JC3IEDM is an emerging standard data model for military command and control developed the under auspices of the Multilateral Interoperability Programme (MIP). The mission of the MIP is to enable interoperability and advance digitization within NATO to support multinational, combined and joint operations [5]. The purpose of the JC3IEDM is to model the information that commanders need to exchange for land-based combat operations.

[1] Including its predecessor, the C2IEDM

The overall JC3IEDM is composed of three data models: Conceptual, Logical, and Physical. The Conceptual Data Model supports general concepts such as actions, organizations, materiel, personnel, features, facilities, locations, etc. The Logical Data Model decomposes (via entity-relationship diagrams) the high level concepts into specific information that is regularly used at the staff level. The Physical Data Model provides the specifications that define the corresponding database schema.

In short, the JC3IEDM schema can describe virtually any land battlefield entity (object-item), condition, task, or relationship. As an example, a unit (which is a type of organization) can have links to action tasks, other organizations (through associations), locations, capabilities, resources, and objectives—to name a few. Likewise, an RDI within a JC3IEDM database can have links to other tables that based on the information contained within the report. These linkages may be to action task event details, status of ongoing tasks, object item capabilities, holding transfer, and associations. This characteristic makes the JC3IEDM a treasure trove of contextual information relating an RDI to a UOI.

**TIPS Queries**

The JC3IEDM is central to the TIPS concept, because the system prioritizes messages not by analyzing their content directly, but by evaluating the database linkages generated as a result of the message's content. In order to prioritize tactical messages for units, TIPS performs a series of in-depth queries against a JC3IEDM database. The purpose of these queries is to characterize the nature of the RDI, as well as how it may be related to the UOI. These queries fall into three categories that are relevant for the prioritization of the message; these are:

1. The quality of the reported data item.
2. The subject of the reported data item.
3. The relationship between the RDI and the UOI.

The determination of the Reported Data Item's quality is based on factors such as its freshness (time since initial report), its accuracy, the type of data, the collection means, the source, and its perceived credibility. Most of these items are actually part of the table entry for each RDI in the JC3IEDM database. While some of these factors are objective (such as the message source and time of receipt), others are more subjective. For example, the credibility rating of the message assumes some input from a human analyst. Further complicating the problem is that much this information may not exist when the message is first reported. Even when the fields are filled in, there is a possibility that they will change over time, as the tactical situation unfolds.

Every message has at least one subject theme. In order to determine what these might be, queries are performed to find the links between the RDI and selected JC3IEDM tables. Each of these links provides a distinct clue into the themes embedded in each message. For example, a linkage from the RDI entry to an entry in Action Task Status table indicates that the RDI communicates a theme about an ongoing action, as well as, its status. Likewise a link to the Object-Item-Capability table implies a theme about an operational capability which, in turn, may support a related theme (operations requirement). A mapping of RDI linkages to subject themes is given in Table 1. Once again, TIPS only mines existing database linkages. The assumption is that these linkages are inserted into the JC3IEDM instance either by human operators or by automated C2 applications.

**Table 1 – Mapping of RDI Linkages to Message Themes**

| JC3IEDM Table Entry | Subject Themes |
|---|---|
| Action-Event | Action |
| Action-Event-Detail | Action |
| Action-Event-Status | Action, Status |
| Action-Location | Action, Location |
| Action-Task-Status | Action, Status |
| Object-Item-Affiliation** | Affiliation → Status |
| Object-Item-Association-Status | Association → Status |
| Object-Item-Capability | Capability → Ops-Requirement |
| Object-Item-Hostility-Status | Affiliation → Status |
| Object-Item-Location | Location |
| Object-Item-Status | Status |
| Organisation-Structure | Association → Status |
| Holding | Resource → Ops-Requirement |
| Target-Personnel-Protection | Resource → Ops-Requirement, Action |

As the embedded themes are identified, the related object-item subjects of the message are retained. An object-item is considered a subject of the message if it is part of a subject theme linkage. For example, if the RDI refers to an action task, that action task is considered a message subject. If the RDI refers to the capability of a hostile unit, that unit is likewise considered a subject. Note that the unit which generated the message is considered a subject by default.

Once the RDI subjects have been identified, the next step is to determine which types of relationships exist between each subject and the UOI. Again, this is accomplished by executing the following queries:

- Shared Action – Is the subject currently engaged in the same action task or event as the UOI? If so, are they direct role (participants) or indirect one (support or observer role)?
- Shared Action Task with Associated Unit – If an Action-Task is shared through an [active] association only, this is an indirect linkage. The bigger the association “chain”, the weaker the linkage.

- Shared Targets – Does the subject and the UOI have shared targets? If so, how many? While shared targets usually imply shared action tasks, this may not always be the case. It is also possible that the subject is a hostile object-item that is a target of the UOI. Shared targets are considered to be direct linkages.
- Shared Context – Within the JC3IEDM, the Context is equivalent to a bulletin board, where assessments of any Battlespace topic can be posted. Object-items may be associated with a Context. This linkage is characterized in terms of true or false only. A shared Context, by itself, indicates an indirect linkage.
- Shared Location – The strength of linkage is proportional to the closeness of the two units. Units may also be associated with common locations (e.g., military base or town), even if they are not currently at those locations.
- Response to Request – Was the RDI generated in response to a request from the UOI? This obviously implies a direct and strong linkage.

For many queries, we are not just interested in a binary yes or no answer. Rather, if a relationship is identified, we try to characterize the strength of the relationship. This is often driven by the number of linkages we find (in many categories, multiple may be found) and if those linkages are direct (subject and UOI directly related) or indirect (related thru a third-party object).

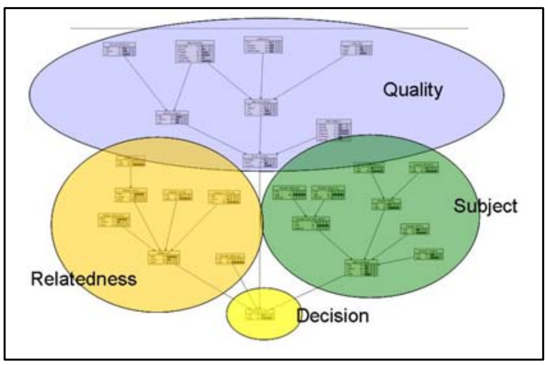
**Bayesian Network**

Once the queries have been executed, the results still need to be rolled up into a single priority. We accomplish this by using a Bayesian Network (BN) to compute the probability that the RDI should be forwarded to the UOI based on the linkages uncovered. BNs are directed acyclic graphs composed of nodes and arcs which model a generalized Probability Distribution Function (PDF) over some domain. The set of nodes within the BN are equivalent to random variables that define the domain. Each variable may be discrete (having a finite number of countable states) or continuous. The arrows represent dependencies between random variables and are sometimes interpreted as causal relationships. Within each node, a local PDF is defined; this encodes the conditional probabilities for that nodes based on the state of parent nodes. For each node, its PDF defines how its state is conditionally affected by the state of the parent nodes; the source of this belief can be either expert knowledge or probabilities learned from data. If a node has no incoming arcs, its PDF defines the apriori likelihood of each state. When evidence is inserted into its nodes, the BN computes the conditional probabilities for the remaining nodes (for which no direct evidence exists). As such, the BN is an extremely useful tool for reaching a decision given what we expect and know to be true within some domain. A more detailed tutorial on BNs can be found in [6].

The findings generated by the linkage test provide the raw data for the BN to interpret. The results of each query are injected as evidence into the nodes of a specially designed BN shown in Figure 1. The figure shows that the BN is segmented into sections which roughly correspond to the three query categories described above. The quality nodes characterize the overall quality of the message. The subject themes located by the queries indicate the overall criticality of the message. The relatedness nodes denote the degree to which the RDI is related to the UOI. Each distinct section of the BN culminates in a "voter" node which provides input to a decision node at the bottom of the network. This decision node (labeled "Forward to Unit") outputs the probability that the message should be forwarded to the UOI; this probability serves as a priority.

## TIPS Architecture

The component-level TIPS architecture is displayed in Figure 2. The TIPS GUI enables users to set up TIPS to process a given JC3IEDM database over a period of time. During an engagement, the database will constantly update to reflect the current tactical situation. This means linkages will change and new reports will continually be added. Because the database changes over time, new messages are initially prioritized and the priority of older messages must be periodically revisited. This is an important point as you would expect the number of linkages associated with an RDI to increase and change over time. Eventually, the priority of the message will stabilize, but this may take a number of hours. Within the architecture, the TIPS Engine regulates the processing of messages within the database.



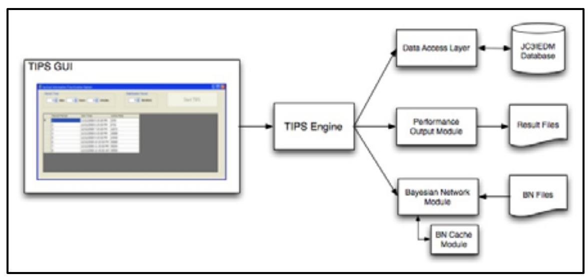




**Figure 1 (left) – TIPS Bayesian Network Model**
**Figure 2 (right) – TIPS Architecture**

The current version of TIPS processes three types of messages: Intelligence messages and Situational Reports (SITREPs), and Obstacle/Weather reports. Each message type has a slightly different set of queries and BN structure (though both are very similar in structure to what is shown in Figure 1). The proper BN must be applied to the right message type. Lastly, we have instrumented TIPS to output metrics in order to gauge its performance and scalability as it processes the JC3IEDM database over time. The results from some of these experiments are presented in the next section.

## 3. Experimentation

Last year, we were tasked by Air Force Research Laboratory (AFRL) to evaluate the scalability of TIPS. This section summarizes our methodology for this task and the results generated to date.

### Previous Experimentation

Previously, we had evaluated the relevance of the recommendations output by TIPS [7]. This evaluation was performed by recording the priorities assigned by TIPS to messages for a fictional operational scenario with those assigned by human analysts. Overall, the results showed that TIPS was able to closely approximate the priorities assigned by human analysts. We found that the difference in assigned priorities between TIPS and human analysts had a mean of 11.8% and a standard deviation of 8.05%. When we placed the priorities in discrete bins, the mean difference dropped to 2.91%. It is important to note that we did separate the analysts into categories, and the priority assignment differences between TIPS and the more experienced analysts was more pronounced. One interesting thing we found was that there was a significant variation in how analysts prioritized various messages. When there was a high degree of consensus among analysts, the different with the TIPS assigned priorities was the smallest.

### TIPS Simulation Harness

One impediment to our experiment was we lacked JC3IEDM databases of sufficient size and complexity to evaluate the scalability of TIPS. To overcome this obstacle, we developed a TIPS Simulation Harness (TSH). The purpose of the TSH is to automatically generate and execute an operational scenario of variable size. The size of the scenario is determined by a number of input parameters, such as scenario duration and number of [friendly] brigades. Once these nominal parameters have been entered, the TSH automatically creates the scenario framework (organizational structures, locations, roads, object-types, etc.) based upon the parameters supplied. TSH then instantiates a JC3IEDM database which is populated with the initialization data.

The next step is to run the simulation in order to generate the message traffic content and associated linkages for TIPS to process. During the simulation run, Action-Tasks automatically planned and tasking messages are issue to participating units. When the Action-Tasks are taking place, participating units generate messages related to their task, unit-type, and location. Generated messages affect the outcome of future messages (i.e., if a unit makes a request for information, another unit must respond within the desired timeframe). Once a report has been generated, it is inserted as a distributed set of table entries in the JC3IEDM database. The specific tables affected are determined by the report types. Ultimately, information associated with a given report has links to entries in the RDI data table. A large part of the effort here is to make the generated scenario as coherent and realistic as possible.

**TIPS Performance Metrics**

The TSH incrementally executes the scenario in discrete slices of time (nominally one hour); these intervals are called revisit periods. During the revisit period, the message traffic will be generated for that scenario period. After all messages are generated for the revisit period, the TSH will notify TIPS, which can begin processing the current JC3IEDM database. When TIPS runs, it processes the new messages, as well as older messages whose priority has yet to stabilize. A fuller description of the TIPS execution parameters can be found in Table 2. After TIPS is finished, the TSH is synchronized to start on the next Scenario interval.

| **Table 2 – TIPS Executive Parameters** | |
|---|---|
| **Parameter** | **Definition** |
| Revisit Period | The amount of time between that must pass before TIPS will process RDIs in the JC3IEDM database. The default revisit period is one sixty (60) minutes. |
| Stabilization Period | The number of revisit periods that must pass without a change in TIPS priority score before an RDI has stabilized. The default value is five (5). Once an RDI has stabilized, it is no longer considered for transmission to a given unit. Stabilization status is determined for each RDI/Unit pair. |

As TIPS executes, we have instrumented the code to output a variety of metrics summarized in Table 3. These metrics are written out to a spreadsheet file every revisit period. These metrics enable analysis of the performance and scalability of TIPS in relation to the complexity of the scenario and the growth of the JC3IEDM database.

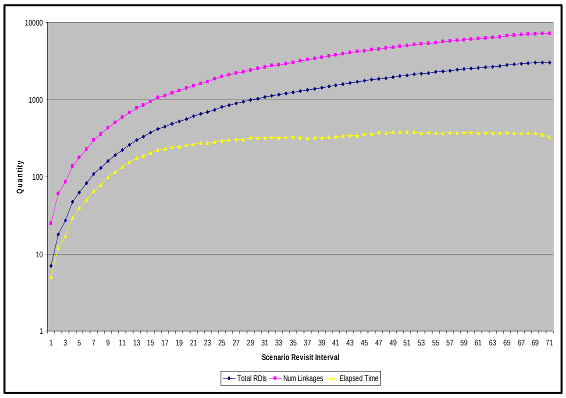
| **Table 3 – TIPS Performance Metrics** | |
|---|---|
| **Metric** | **Definition** |
| Num RDIs | Number of Reported Data Items (RDIs) inserted into the JC3IEDM database instance by the TSH since the start of the scenario |
| Num Linkages | Number of first-order linkages (references to RDI as a foreign key) inserted into the JC3IEDM database. |
| RDI/Unit Pairs | The total number of TIPS queries performed per RDI. We track two items in this category:<br>• Active – still being evaluated<br>• Stabilized – evaluation has ceased |
| Elapsed TIPS CPU time (per revisit period) | The amount of elapsed CPU time to execute all TIPS queries during a given revisit period. |
| RDI score change | The difference in the TIPS priority for a given RDI. The difference in computed using the average score for the first revisit period (when the RDI becomes active) and the last (when it becomes inactive). The MEAN and STDEV statistics are computed for this metric. |

| | |
|---|---|
| RDI time active | The average number of revisit periods that an RDI is active. This is computed over all RDIs each revisit period. The MEAN and STDEV statistics are computed for this metric. |

**Run-time Experiment Results**

One of our design objectives was for the run-time performance of TIPS to scale linearly as the number of units increased. To test this, we ran TIPS on a number of different sized scenarios; in terms of units, the scenario sizes were: 74, 133, 185, 295, and 583. Figures 3 and 4 show the intra-scenario scalability for two scenarios: 185 units and 295 units, respectively. During the course of the 72 hours scenario, the size of the database grew as more messages are generated. In both cases, the results show that the elapsed CPU time per revisit interval holds stable despite the steady increase in database size. One of the most common measures of run-time performance is CPU reserve. In terms of this metric, TIPS utilized only 22% of the available 3600 seconds in the revisit period for the larger scenario.

The principal reason why the CPU utilization holds steady is that RDIs are no longer processed by TIPS once their priority has stabilized. Figure 5 illustrates this point: as the total number of RDIs in the database increase, those under active consideration level off early in the scenario. Of course, some fluctuation in this regard is to be expected. Figure 6 shows the mean and standard deviation of RDI lifetimes as the scenario progresses. For the scenarios shown, the average RDI lifetime is 8.39 hours, with a peak life of 11.1 hours; the 295 unit scenario had comparable results, with the 583-unit scenario showing a slight decrease. While the mean RDI lifetime grows incrementally, this gradual increase is not large enough to significantly affect the run-time performance.



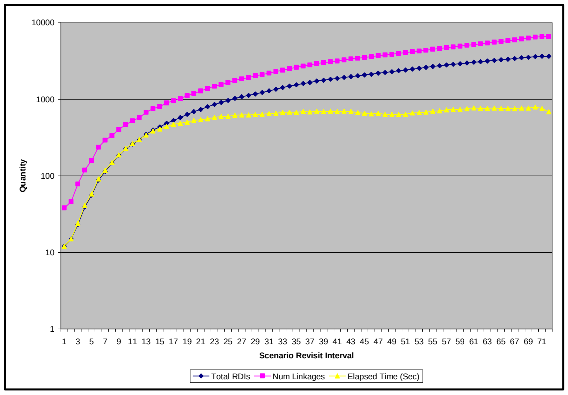
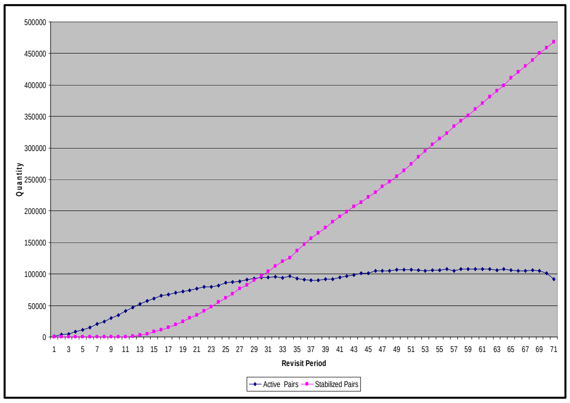




**Figure 3 (left) – Run-time performance for 185 unit scenario**
**Figure 4 (right) – Run-time performance or 295 unit scenario**



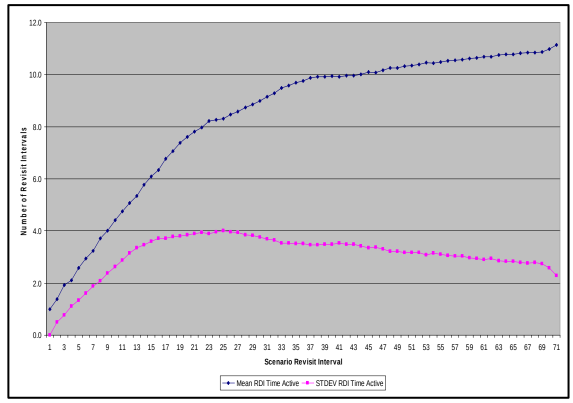
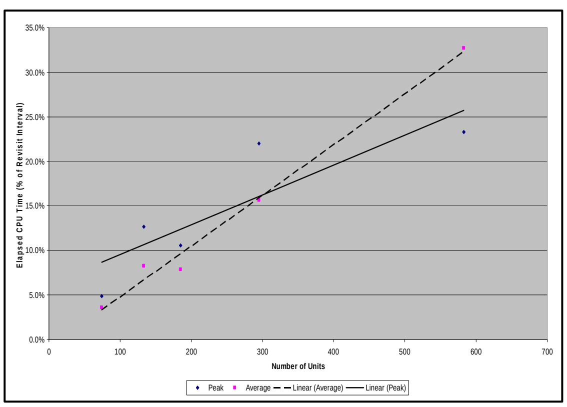




**Figure 5 (left) – RDI stabilization rate for 185 unit scenario**
**Figure 6 (right) – RDI lifetime for 185 unit scenario**

Of course, the above figures only portray the intra-scenario run-time performance for TIPS. We characterize inter-scenario performance by evaluating a range of scenario sizes. Figure 7 shows the percentage of CPU utilization on a per revisit period basis as we increase the scenario size. Both peak and average CPU utilization are shown. In both cases, the growth is roughly linear. Likewise, Figure 8 shows a linear growth in processing time (on a per linkage basis) as the number of units increases.

The results of these experiments is significant, because it shows the TIPS approach is linearly scalable for the type of databases that would be generated by a large coalition operation. The size of the scenario and its duration also serve to stress-test the system. Each scenario simulates a 72 hour period—the mean and peak performance measurements were collected over this duration.



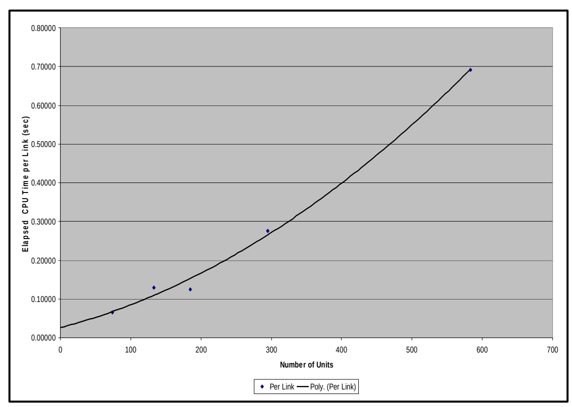




**Figure 7 (left) – Scalability based on revisit interval utilization**
**Figure 8 (right) – Scalability based on per link processing**

## 4. Summary

TIPS extracts context from JC3IEDM databases to perform automated prioritization of military messages. Though it is automated, we do not recommend TIPS as a replacement for military analysts; instead, we see it as an electronic assistant to cue them to potentially important information. The primary motivation for the TIPS approach is the potential to help combat information overload within C2 centers. Consider the effort required for a human C2 analyst to quickly prioritize the importance of every report for each unit. The difficulty of this task is compounded by the dynamic nature of warfare—facts are continually being added or changed. This means that a report that seemed innocuous might in fact be important due to subtle (second or third order) relationships not readily apparent to an overworked analyst. Further, consider that a less-than-fresh report might become more important to a unit over time, as additional facts become known. Overcoming information overload to maintain situational awareness is critical. Losing SA can result in friendly fire tragedies, such as the accidental downing of two Black Hawk helicopters during Operation Provide Comfort in 1994.

This paper provided a conceptual explanation of TIPS and tested the scalability of TIPS with a variety of scenario-generated JC3IEDM databases. Within a scenario, the TIPS CPU utilization (per revisit period) tends to level off early in the scenario. This is a favorable result and is largely dependent on the phenomenon that most messages will stabilize in terms of their priority within about 8 hours. As a result, even though the number of raw messages increases, the TIPS processing load levels off. The inter-scenario results show that TIPS scales linearly as the size of the database increases. This trend is likewise encouraging as it indicates that TIPS processing overhead is manageable in an operational setting. As a result, the required computing resources can be sized apriori to avoid delays in the processing of critical messages.

## References


[1] Grishman, R. and Sundheim, B. 1996. “Message Understanding Conference-6: a brief history”. Proceedings of the 16th Conference on Computational Linguistics - Volume 1 (Copenhagen, Denmark, August 05 - 09, 1996). International Conference On Computational Linguistics. Association for Computational Linguistics, Morristown, NJ, 466-471.

[2] Sundheim, B. M. 1995. “Overview of results of the MUC-6 evaluation”, Proceedings of the 6th Conference on Message Understanding (Columbia, Maryland, November 06 - 08, 1995). Message Understanding Conference. Association for Computational Linguistics, Morristown, NJ, 13-31. DOI=http://dx.doi.org/10.3115/1072399.1072402

[3] Angus, R.G. and Heslegrave, R.L. 1985. “Effects of Sleep Loss on a Sustained Cognitive Performance during a Command and Control Simulation”, Behaviour Research Methods, Instruments, and Computers, 17(1), 55-67.

[4] Thomas, T. L. (2000) "Kosovo and the Myth of Information Superiority", Parameters: US Army War College, Spring 2000, Volume 30, Issue 1.

[5] Joint Consultation, Command and Control Information Exchange Data Model, (JC3IEDM); http://www.mip-site.org/

[6] Charniak E. 1991. "Bayesian Networks Without Tears", AI Magazine, Volume 12, Number 4, 50-63.

[7] Marmelstein, R.E., Devito, M.G., Linderman, M.H. et all. (2008). "Link Analysis for Tactical Information Prioritization", Proceedings of the Ninth International Conference on Data Mining, Cadiz, Spain, May 2008.