Editorial

ALTHOUGH the Artificial Intelligence area is almost as old as Computer Science it is a continuous source of large amounts of new developments, and it is still growing since new related fields are emerging and quickly consolidating, making AI stronger every passing day. As a consequence there is an intense amount of excellent works coming out from laboratories deserving to be known by a wider and diverse specialized audience. With this information in mind, and as a modest collaboration to the expansion of AI, *Polibits* has selected a set of important papers reporting works which are published in this issue. The selected works, besides showing the large variety of new developments and potential applications, reflect the diversity of traditional and emergent AI fields which are in constant production of new valuable findings. A brief description of each of these works is given next to illustrate the contents to be found in them.

In the article entitled "*LG-PACKAGE: New Frontier*", Boris Stilman and collaborators report the development of LG-PACKAGE, which is a set of the Linguistic Geometry (LG) tools. In the paper it is stated that LG is a type of game theory that generates best strategies for all sides in a conflict in real time. It also describes the main advanced features of LG-PACKAGE, converting LG-PACKAGE into software of industrial strength applicable to a wide scope of defense systems. The authors also state that US and British defense agencies and the world major defense contractors utilize these tools.

David Sundgren presents "*Expected Utility from Multinomial Second-order Probability Distributions*", in which establishes that computing the expected utility of a decision alternative it may not always be possible to give precise values for the utilities and probabilities of the possible outcomes. As a possible solution the author considers the problem of maximizing expected utility when utilities and probabilities are given by discrete probability distributions so that expected utility is a discrete stochastic variable. He also suggests that a decision rule that reflects the uncertainty present in distribution-based probabilities and utilities- In the article an example of this rule in action is shown with multinomial second-order distributions. And it is demonstrated that discrete second-order probability distributions allow for updating through observations in a way that continuous distributions would not.

A new content-based method for the evaluation of text summarization systems without human model summaries which is used to produce system rankings is shown in "*Summary Evaluation with and without References*" by Juan-Manuel Torres-Moreno *et al*. There they apply a comparison framework to various well-established content-based evaluation measures in text summarization such as COVERAGE, RESPONSIVENESS, PYRAMIDS and ROUGE studying their associations in various text summarization tasks including generic multi-document summarization in English and French, focus-based multi-document summarization in English and generic single-document summarization in French and Spanish.

In "*Creation and Usage of Project Ontology in Development of Software Intensive Systems*", P. Sosnin shows that the key problem of successful developing of software intensive systems (SIS) is adequate conceptual interactions of designers in the early stages of designing. And that the success of the development can be increased with the use of a project ontology the creation of which is being embedded into the processes of the conceptual solving of the project tasks and specifying the project solutions. The author states that the essence of the conceptual design is a specification of conceptualization, and claims that the main suggestion of this article is the creation of the project ontology in the form of a specialized SIS which supports the conceptual activity of designers.

In his paper "*The Role of Automation in Instruction*" Joseph M. Scandura demonstrates the benefits of application of computers in e-learning and presents a novel conceptual scheme of this application.

A description of some important ideas of how to understand and extract the mental representation of individuals in the decision making and planning of trips, related to daily travels is given in the paper entitled "*A Revision and Experience using Cognitive Mapping and Knowledge Engineering in Travel Behavior Sciences*" by Maikel León and collaborators. The reason given is that this is useful information that can be used in transport demand prediction, analysis and studies**.**

In the paper entitled "*Mixing Theory of Retroviruses and Genetic Algorithm to Build a New Nature-Inspired Meta-Heuristic for Real-Parameter Function Optimization Problems*", Renato Simões, Otávio Noura Teixeira, and Roberto C. Limão describe the development of a new hybrid meta-heuristic for optimization based on a viral lifecycle, focusing in the retroviruses, called Retroviral Iterative Genetic Algorithm (RIGA). This algorithm uses Genetics Algorithms (GA) structures with features of retroviral replication, providing a great genetic diversity, confirmed by better results achieved by RIGA comparing with GA applied to some Real-Valued Benchmarking Functions.

To complement the selected set, in "*Swarm Filtering Procedure and Application to MRI Mammography*" Horia Mihail Teodorescu and David J. Malan investigate the use of biologically-inspired swarm methods for signal filtering. It is stated that the signal is modeled by the trajectory of an agent playing the role of the prey for a swarm of hunting agents. The swarm hunting the prey is the system performing the signal processing. The movement of the center of mass of the swarm represents the filtered signal. The position of the center of mass of the swarm during the virtual hunt is reverted into grayscale values and represents the output signal. They also show some results of applying the swarm-based signal processing method to MRI mammographs.

The paper of Marvin Arias "*Analysis of Multipath Propagation based on Cluster Channel Modelling Approach*" is an interesting application of AI techniques in the telecommunication field.

The paper "*Formalization of Basic Semiotic Notions in Set Theoretic Terms*" by Alisa Zhila introduces a reader in the complex world of semiotic concepts and presents a formal model of basic notions of semiotics using notions of logic.

With the publication of the chosen papers the editor of this issue of *Polibits* is sure that its reading will bring some new and different insights in several AI research fields as to suit all kinds of scientific preferences in the area. I hope the reader will enjoy and benefit from the presented papers.

Carlos Alberto Reyes-García
Research Professor,
National Institute for Optics, Electronics
and Astrophysics (INAOE),
President of the Mexican Association
of Artificial Intelligence (SMIA),
Mexico

LG-PACKAGE: New Frontier

Boris Stilman, Vladimir Yakhnis, Oleg Umanskiy, Ron Boyd, Viacheslav Pugachev, and Lance Hagen

***Abstract*—This paper describes "new frontier" reached in the development of LG-PACKAGE, a set of the Linguistic Geometry (LG) tools, introduced first in 2004. LG is a type of game theory that generates best strategies for all sides in a conflict in real time. The paper describes the main advanced features of the versions (through Version 3.7) of LG-PACKAGE released gradually from 2004 through 2010. These releases converted LG-PACKAGE into the software of industrial strength applicable to the wide scope of defense systems. The US and British defense agencies and the world major defense contractors utilize these tools.**

***Index Terms*—Linguistic Geometry, Game Theory, Modeling and Simulation, Search Problems**

I. Introduction

LINGUISTIC Geometry (LG), a type of game theory, was first introduced in [4] in 1992. A much deeper account in LG is provided in [5]. Future directions of development of LG are considered in [10].

LG-based tools automatically generate winning strategies, tactics, and courses of action (COA) and permit the warfighter to take advantage thereof for mission planning and execution. LG looks far into the future – it is "predictive". With unmatched scalability, LG provides a faithful model of an intelligent enemy and a unified conceptual model of joint military operations. The LG tools are based on the concept of the LG *hypergame*. A hypergame is a system of several abstract board games (ABG) of various resolutions and time frames. The games are "hyper-linked", whereby a move in one of the games may (or may not) change the state of the rest of the games included in the hypergame. More details about the foundations and applications of LG are given in [3] - [10].

A set of the LG software tools, LG-PACKAGE [3], includes the following six generic components: GDK (Game Development Kit), GIK (Game Integration Kit), GRT (Game Resource Tool), GST (Game Solving Tool), GNS (Game Network Services) and GMI (Game Mobile Interface). Game construction layer includes GDK, game solving layer includes GRT and GST, game service layer that includes GIK, GNS and GMI supports both game construction and game solving.

Game Development Kit (GDK) permits creation of battlespaces, missions, and campaigns. With GDK, the analysts may optionally develop domains (Air, Ground, Joint Operations, etc.) from which specific campaigns and missions may be developed with a significant level of automation. The domain development includes modeling military hardware (UAV, manned aircraft, tanks, SAM, ships, etc.) as LG piece-templates and automatic generation of battlespace/theater templates from elevation maps in the form of DTED and shape files. Existing and future (conceptual) military systems and their concept of operations can also be modeled.

Game Integration Kit (GIK) permits integration of LG-PACKAGE into a federation of other tools, such as military C2 (Command and Control) systems (e.g., FBCB2, DCGS-A, CPOF), intelligence databases, external synthetic environments and SAF (Semi-Automated Force) simulators, control theory based tools like hybrid systems and discrete event systems, stochastic modeling tools, knowledge-based tools, etc. GIK allows LG tools to operate as a back-end to any other system – receiving all needed input data from and sending computed COA to an existing system. It further allows LG-PACKAGE to generate enhanced strategies employing access to additional information such as historical databases or real-time sensor and positional data. GIK has already been used for integration with several systems: FBCB2, JVMF, DCGS-A, OneSAF (OTB), TotalDomain, InterScope, FLAMES, JSAF, VR-Forces, and others. GIK supports a variety of communication interfaces – publish/subscribe or direct socket connections. Using GIK each LG-PACKAGE component can function as either a client or a server to provide more flexibility for integration. Both XML and binary messages are supported. The XML formats are strictly documented using XML Schema Definition (XSD) files and provide a straightforward integration method. The binary format delivers a much smaller message size and provides the greatest benefit in low bandwidth situations.

Game Resource Tool (GRT) determines the start state of the game, i.e., resources needed for a side at the start of the game in order to win. It provides an optimal (or near optimal) resource allocation for a given player (side) for every gaming template within the domain where the resources for all the

Manuscript received August 17, 2010. Manuscript accepted for publication September 12, 2010.

This work was supported in part by STILMAN Advanced Strategies, LLC under internal R&D. Constant development and improvement of LG-PACKAGE was also supported by the licenses purchased by the US Department of Defense, the UK Ministry of Defence, The Boeing Corp. (USA), BAE Systems (UK), SELEX Galileo (UK, a Finmeccanica company) and others.

B. Stilman is with STILMAN Advanced Strategies, 3801 E. Florida Ave., Suite 400, Denver, CO 80210 and the Department of Computer Science and Engineering, Campus Box 109, University of Colorado Denver, Denver, CO 8017-3364, USA (303-717-2110; boris@stilman-strategies.com).

V. Yakhnis is with STILMAN Advanced Strategies, 3801 E. Florida Ave., Suite 400, Denver, CO 80210, USA (vlad@stilman-strategies.com).

O. Umanskiy is with STILMAN Advanced Strategies, 3801 E. Florida Ave., Suite 400, Denver, CO 80210, USA and the Department of Computer Science and Engineering, Campus Box 109, University of Colorado Denver, Denver, CO 8017-3364, USA (oleg@stilman-strategies.com).

R. Boyd is with STILMAN Advanced Strategies, 3801 E. Florida Ave., Suite 400, Denver, CO 80210, USA (ron@stilman-strategies.com).

V. Pugachev is with STILMAN Advanced Strategies, 3801 E. Florida Ave., Suite 400, Denver, CO 80210, USA and the Department of Computer Science and Engineering, Campus Box 109, University of Colorado Denver, Denver, CO 8017-3364, USA (slava@stilman-strategies.com).

L. Hagan is with STILMAN Advanced Strategies, 3801 E. Florida Ave., Suite 400, Denver, CO 80210, USA (lance@stilman-strategies.com).

other players are already specified. While allocating resources so that the designated side may fulfill its goals with a given overall probability of success, GRT minimizes the total "opportunity cost" of the resources.

Game Solving Tool (GST) is the key component of LG-PACKAGE. It predicts and simulates the engagement beginning from the start state

– selected manually,
– received from other software tools via GIK, or
– generated by GRT.

The engagement is executed by placing and moving the pieces on the board and by automatically, in real time, making decisions for one or more sides in a conflict. GST generates the best strategies, tactics, and COA for every battlespace within the domain. To provide various levels of automation, GST can be executed in several modes, automatic, interactive, and monitoring.

Game Network Services (GNS) support automatic, parallel and distributed execution of multiple components of LG-PACKAGE over the network of computers including local high-speed networks, Internet, or combinations of both. GNS support concurrent distributed construction and execution of the large-scale LG hypergames. GNS provide extreme robustness to the LG hypergame, so that various adverse hardware/software events (anywhere in the network) would not interrupt hypergame execution. In the worst case, they may reduce execution speed.

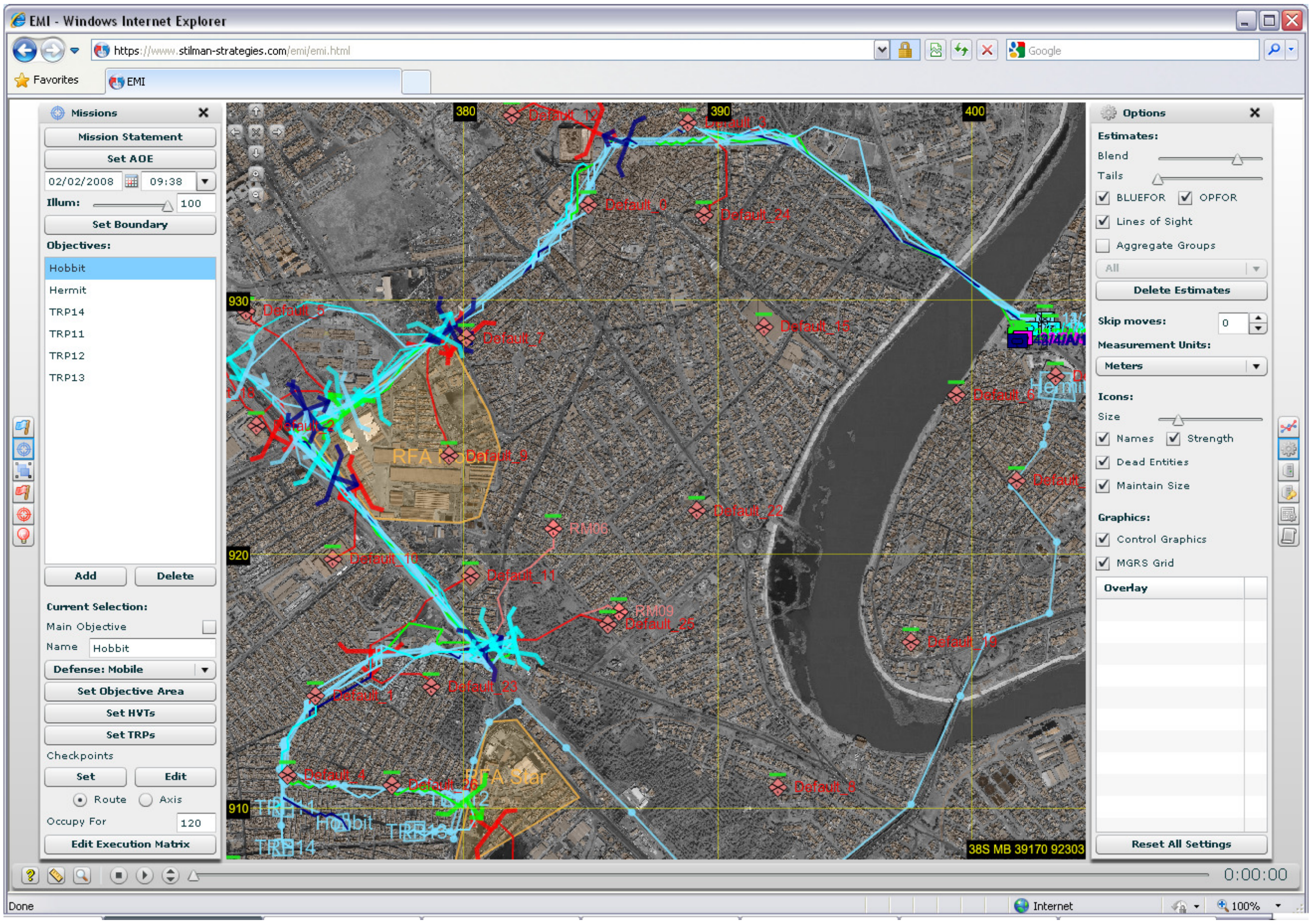
Game Mobile Interface (GMI) (Fig. 1 and Fig. 2) delivers over any network (including wireless networks and Internet) a modern, simple and task-customized interface to a particular application of LG-PACKAGE. It provides the user with an easy and natural interface to set up specific scenarios that are of the highest interest to him/her, without cluttering the interface with overwhelming options not needed for the specific intended use cases. GMI can then visualize - and let the user manipulate - the LG construction and computational components (GDK, GST, GRT and any additional data) in a similarly customized and natural manner for the desired user tasks. GMI can be executed from within any standard web browser without installing any additional software and thus makes power of LG-based COA computations easily available to any user with Internet or local network access. Different customizations of GMI include a version for touch screen computers and a different version for handheld devices such as PDAs or cell phones.

The original LG-PACKAGE, first released in 2004, consisted of just three components, GDK, GRT and GST. The subsequent major releases of LG-PACKAGE included other generic components. The basic features of all the generic components are described in [3] and [9]. In this paper we describe the advanced features that were introduced in 2004 - 2010.

II. Realistic Sensors and Communications

Realistic Sensors. LG-PACKAGE allows the user to introduce into simulation a wide range of realistic sensor types. In particular, currently simulated sensors can provide partial information about enemy objects. Depending on user-defined sensor parameters, when an enemy object comes into the detection range during a simulation, the friendly force is able to determine a combination of the following four basic parameters (or attributes) of the object: location, affiliation, type, and armament. Various settings of the parameters cover all the feasible combinations. In addition, the user can create custom sensor types. For instance, laser guidance, visual confirmation, and fire control radar could all be added as custom sensor types and later used to define guidance requirements for weapon platforms. Similarly, "Detected", "Tracked", "Recognized", and "Identified" could also be specified as custom detection types and later used to define ROE (Rules of Engagement) for missions. Using these features, the user can specify various types of guided weapons (e.g., "laser"- and "radar"-guided weapons) and their guidance sensors, as well as specify missions with restrictive ROE - such as allowing targets to be prosecuted only if they have been "identified" by an appropriate sensor. The user can specify which detection states can be reached by this sensor against each of the defined object types. For the sensors simulated by LG-PACKAGE, the user can introduce Pd (Probability of Detection) functions. This introduction can be made for each sensor-detection state-platform combination. Powerful GUI provides convenient means to easily introduce functions of any shape and complexity. For example, a sensor could be defined to provide location of certain types of enemy aircraft with Pd = 100% up to 20 km range, and slowly drop to 0% by the range of 50 km. The shape of this Pd function can easily be defined by the user. The definition of this same sensor could be extended to include the following. This sensor would be able to detect the type of enemy objects with 75% probability at 10 km range and 0% probability beyond that range. Other target types could be specified as invisible to this particular sensor.

Sophisticated Sensor and Worldview Models. LG-PACKAGE includes an sophisticated model of sensor interactions. Instead of having a Boolean parameter describing each entity, i.e., either known or not known in a particular worldview, LG-PACKAGE employs a parameter with a 'certainty' value assigned to each entity. When the entity is originally detected by someone's sensor, it is known with full certainty, which entity was detected. Then, this information decays overtime indicating that this knowledge is outdated. The speed of such decay is dictated by the properties of the entity, as its maximum speed. A subsequent sensor contact would restore the certainty back to 100%. Another concept utilized in LG-PACKAGE is called a "negative sensor contact". When a sensor is used to scan a location where the entity was last known to be, and yet it is not currently detected there, the certainty value of that entity is rapidly lowered based on the probability of detection of the sensor. The negative sensors can be used to confirm that the entity is not where it was believed to be. Furthermore, "intelligence entities" can be introduced into worldviews to improve the ability to setup scenarios with incomplete and uncertain



EMI - Windows Internet Explorer

https://www.stilman-strategies.com/emi/emi.html

Missions

02/02/2008 09:38

Illum: 100

Objectives:
Hobbit
Hermit
TRP14
TRP11
TRP12
TRP13

Current Selection:
Main Objective □
Name Hobbit
Defense: Mobile

Checkpoints
● Route ○ Axis
Occupy For 120

Options

Estimates:
Blend
Tails
☑ BLUEFOR ☑ OPFOR
☑ Lines of Sight
□ Aggregate Groups
All

Skip moves: 0
Measurement Units: Meters

Icons:
Size
☑ Names ☑ Strength
☑ Dead Entities
☑ Maintain Size

Graphics:
☑ Control Graphics
☑ MGRS Grid

Overlay

0:00:00

Done

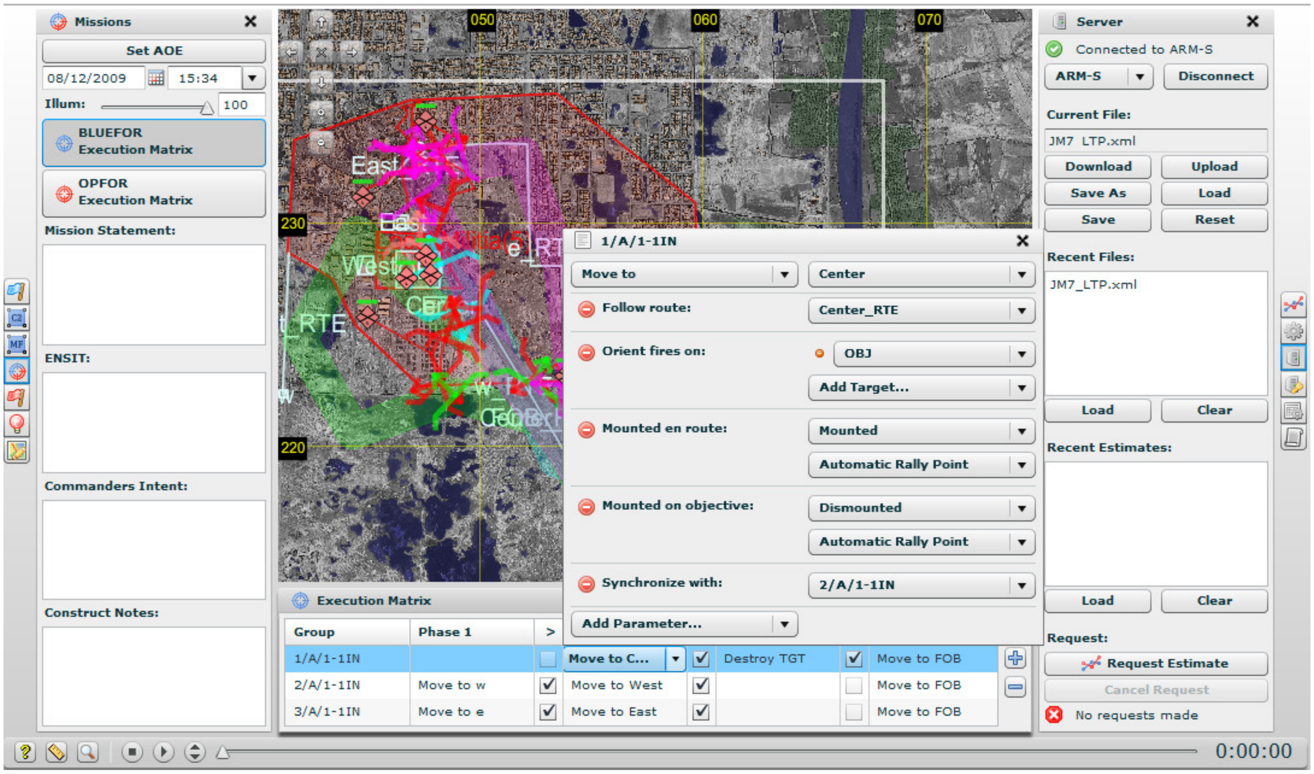
Fig. 1. GMI: Detailed COA for Blue and Red forces in complex urban terrain.

information. This permits modeling a wide range of missions and behaviors, including Search missions and Intelligence Verification missions (Section III). For instance, a UAV can be tasked to fly to all locations with intelligence pieces and verify whether there is (or is not) an actual enemy entity there employing positive or negative sensor contacts.

Realistic Communications. LG-PACKAGE allows modeling realistic "imperfect" communications. It allows the user to break down each of the conflicting sides into communication groups. Each of the communication groups maintains its own worldview and uses an independent LG Engine to generate strategies, COA and movement for all its members. The user can also define communication links and their associated delays. For each communication group, the associated LG Engine bases its reasoning only on information available within the communication group's worldview. This information is fused from the sensor inputs from all the entities of the communication group, as well as from information arriving through communication links to the other communication groups (with appropriate communication delays as applicable). In addition, LG-PACKAGE allows the user to simulate and assess the dependencies of outcomes of various engagements upon the communication infrastructures. Various communication delays between the communication groups, breakdowns of the forces into communication groups, as well as dynamic real-time changes to the communication network can be experimented with to analyze their effect on the simulation. LG-PACKAGE automatically enables the information flow from one communication group to another via the shortest path through any allowed communication links and nodes. This flow can change dynamically with changes in the communication infrastructure, e.g., if an important intermediate node is destroyed in the engagement. Furthermore, communication groups allow experimentation with the effects of appropriate command structures (Section V) upon the outcome of engagements by modeling the improved information flow stemming from an efficient command hierarchy. Finally, the GUI allows the user to visualize the worldview of each individual communication group to understand the differences in their current operational picture and their impact on the groups' decision making, i.e., computation of strategies and COA.

III. Complex Missions and Operations

Mission Editor. GST includes a highly flexible Mission Editor. Communication groups described above (Section II) can further be broken into task groups, which can be assigned missions via the Mission Editor. Each mission can be assigned to multiple task groups to be performed cooperatively or to allow LG to choose the best fitting task group for the mission. At this time, Attack, Defend, and Relocate mission types are



Missions

08/12/2009 15:34

Illum: 100

BLUEFOR Execution Matrix

OPFOR Execution Matrix

Mission Statement:

ENSIT:

Commanders Intent:

Construct Notes:

1/A/1-1IN

Move to: Center

Follow route: Center_RTE

Orient fires on: OBJ

Add Target...

Mounted en route: Mounted

Automatic Rally Point

Mounted on objective: Dismounted

Automatic Rally Point

Synchronize with: 2/A/1-1IN

Add Parameter...

Execution Matrix

Group	Phase 1	>					
1/A/1-1IN		☐	Move to C...	☑	Destroy TGT	☑	Move to FOB
2/A/1-1IN	Move to w	☑	Move to West	☑		☐	Move to FOB
3/A/1-1IN	Move to e	☑	Move to East	☑		☐	Move to FOB

Server

Connected to ARM-S

ARM-S

Current File: JM7 LTP.xml

Recent Files: JM7_LTP.xml

Recent Estimates:

Request:

No requests made

0:00:00

Fig. 2. GMI: Entering detailed Mission Order in Execution Matrix format modeled after US Army standard.

supported. Missions can target specific units or all units within a specific area that meet the Targeting Criteria. Such criteria can specify, which types of units can be attacked, which sides they must belong to, as well as detection states that must be attained by those units before they can be attacked. Even more complicated Rules of Engagement can be set up using Targeting Criteria based on simulation time, status of other missions, or even friendly or enemy force strengths. The Mission Editor permits employment of the logical expressions using Start, Pass, and Fail Mission Criteria. This allows the user to specify combination of events or parameters that must be met before a mission can start, be considered successful or failed. Each such criterion can be a complex logical proposition of variables that include simulation time ranges, status of other missions, friendly or enemy force strengths, etc. Force strength parameter can further be fine tuned by the user to only include certain types of units, and only the units within certain areas or groups. Missions can also include way points to be passed through on the way to the main objectives. The Mission Editor allows the user to simulate available intelligence on enemy missions by permitting reflected missions, i.e., those to be executed by one side and such that their existence is known to the other side as the other side's "intelligence".

Missions' Hierarchy. Missions in LG-PACKAGE can be organized hierarchically. A mission can contain other missions within it. Different types of such hierarchies are supported. A "sequential" mission group can be used to quickly specify several submissions that need to be executed in order. A "synchronized" mission group controls the sub-missions to be executed and finished altogether. Another type of a mission group is a "segmented" mission that is essentially a single mission that is broken down into smaller components with different actions and tasks. Such groupings facilitate creation of complex interconnected mission structures for scenarios even easier while taking advantage of other existing Mission Editor features.

Execution Matrix. The Mission Editor provides a way to define very complex and flexible mission structures (see above). However, some users require an interface that is structured differently. As part of GMI, LG-PACKAGE includes an additional mission editor, the Execution Matrix (Fig. 2). This method is based on the actual US Army method for specifying mission orders which is a matrix of organization groups along one axis, and the time or mission phases along the other axis. Each cell of the matrix contains a task order that specifies what each group has to do during each phase of the mission. The task order consists of a "task-action" and a "task-target", where the target type depends on the action. For instance, an "attack" action can be applied to both an objective area and to an enemy force, while a "clear" action can only be applied to an objective area or a route. The tasks can also include additional parameters which specify "how" the task is to be done, e.g., waypoints to be followed, orientation of forces on the objective, and whether the mission is to be mounted or dismounted. The Execution Matrix approach does not allow as much flexibility as the general LG Mission Editor (see above); however, it provides an approach, which is familiar to the US Army trained personnel and covers the range of mission orders that they are required to execute. Due to its simplicity, Execution Matrix is also much faster to use.

Search Missions. Currently, a list of mission types includes various Area Search Missions. Such missions are defined in terms of the area to be searched, types of entities being searched for, and desired search pattern – such as "creeping line" or "square patterns". In addition, these search patterns can be automatically computed based on sensor parameters, e.g., probabilities of detection, of the search assets to achieve desired coverage for the search area. The search missions are integrated into the rest of the software functionality; they can be used in conjunction with other missions, and take full advantage of the rest of the COA generation capabilities. For example, the user can model a scenario simulating a search by UAV assets for hostile air defense resources, followed by a more thorough search executed by manned aircraft with fighter escort for high value targets, with the escorts responding to any threats to the search assets, culminating with a time critical targeting (TCT) missions to destroy any discovered high value targets.

Generic Mount/Dismount Missions. LG-PACKAGE includes the ability to model operations that involve units transitioning between mounted and dismounted actions within the same scenario. This allows modeling the following sample operations. A platoon of infantry is traveling to the target area mounted on Infantry Combat Vehicles, dismounting and attacking the enemy on foot with vehicles used for fire support, re-mounting to move to the next objective, and dismounting en-route if a threat is discovered. The system can automatically select mount rally points of several types to support TTP (Tactics, Techniques and Procedures). For instance, a single rally point can be chosen for the entire platoon to assemble and mount together or individual vehicles can pick up their passengers independently with a separate mount rally point for each vehicle. This functionality is not restricted to the Army land operations. For instance, this can be applied to modeling a battleship transporting unmanned attack submarines or other assets such as attack helicopters, deploying those submarines & helicopters in the mission area or defensively as needed, performing the attack jointly, followed by the submarines and helicopters "re-mounting" the battleship and proceeding to the next mission. This functionality can be controlled by the user by specifying desired mounted or dismounted operation for each mission, as well as defining relationships between different entities to specify possible mount options.

Support Missions. LG-PACKAGE permits constructing missions that support other missions. Such missions do not have their own target but are rather assigned to a particular different mission or a group to be supported. Such support roles include:

- "follow and support",
- "quick reaction force", and
- "support by fire".

In a "follow and support" mission, the support group will follow behind the supported group, and as needed, come forward to assist the supported group against any threats or to help attack its final target.

In a "quick reaction force", the support group will remain in its original location and, if any threat to the supported group is detected, rapidly move in to intercept such threat.

"Support by fire" is used typically when a main attack on the final objective is to be assisted by establishing a base of fire on a different axis and suppressing the enemy by direct or indirect fire during the final assault. GST can automatically calculate the appropriate location for such base of fire in a support mission.

Missions for MOUT Operations. LG-PACKAGE provides extensive support for modeling Military Operations in Urban Terrain (MOUT). This is achieved by taking advantage of the cumulative effect of all other features of LG-PACKAGE combined with some advanced functionality for modeling CONOPS (concept of operations), SOP (Standard Operating Procedures) and TTP for urban asymmetric operations. The most important features are as follows:

- Competency and aggressiveness properties that can be assigned to entities to simulate different behaviors, e.g., differentiate between militia and trained foreign fighters,
- Tactics other than direct force-on-force, such as running away, hiding behaviors, non-aggressive posturing,
- Rules of Engagement (ROE) on weapon use, such as not engaging the enemy until the enemy engages first,
- Indirect fire support weapons with complex ROE, such as the size and armament of the target, and proximity of friendly forces,
- Customizable generation of the LG zones [5]-[10] simulating different SOP and TTP, such as non-aggressive posturing, formations en-route, reserve forces, medical evacuation, and reactive defensive tactics,
- Synchronization of platoons to achieve maximum effect of overwhelming force and massing of weapon fires, and
- Maintaining cohesion of a platoon throughout the operation.

IV. Complex Terrain

Complex Terrain Modeling. LG-PACKAGE allows the user to model domains and scenarios involving complex terrain models. This includes terrain elevations, separation of land and water, and an additional layer of the terrain features data including buildings, roads, bridges, rivers, lakes, and forested areas. In addition, a notion of "density" is introduced to distinguish between cells of the game board that are completely occupied by a feature (such as building or canopies) and those that are only partially occupied by this feature. These terrain models are completely integrated into the rest of the LG algorithms. For instance, "flexible" reachability relations [5], [6] can be defined as follows. They can be different for land and for roads. In addition, we can define reachability relations that only apply on water; or those that permit faster movement when moving through forests of lower density, slower - in more dense areas, and even slower - in heavily built-up areas. We can define weapons that can only be fired at targets that are in the open rather than those taking cover in buildings or heavily forested areas. We can define sensors that have different levels of penetration depending on what is encountered along the line of sight from the sensor to the target – whether it is small buildings, lighter or heavier forested areas. This permits a variety of locations and domains to be modeled realistically, e.g., littoral operations, ground operations in rural terrain, as well as operations in urban terrain.

Terrain Analysis. LG-PACKAGE includes a customizable terrain analysis engine that can process complex terrain models including buildings, roads, rivers, lakes, bridges, and canopies. This terrain analysis engine permits to distinguish dangerous and preferred areas based on lines of sight, terrain, range of friendly and hostile weapons, current known or estimated positions of enemy forces. Such analysis can be customized employing an extensive GUI. This GUI permits to produce completely different (and tactically valid) terrain analysis for different types of entities. For instance, the analysis for dismounted troops could be configured to highlight wide open areas within range of weapon fire from built-up areas as the most dangerous, while considering locations in the buildings that are high enough to provide good lines of sight over neighboring areas as the best observation/fire positions. For vehicles, this analysis could be reversed to show that it is most dangerous for vehicles to be tight in between buildings (where they are susceptible to RPG fire), while the best positions are in open areas where the effect of long ranges of fire of their weapons is maximized. This analysis can also be used to indicate user preferences, e.g., traveling through forests or through buildings, over land or over water, high in the air or low to the ground, etc. The results of such terrain analysis are directly applied to affect calculation of COA by influencing the LG trajectories and zones being generated. Thus, all the forces are choosing the safest and most efficient routes to dominate the enemy forces.

Automated Terrain Import. LG-PACKAGE permits developing scenarios for a given geographical location by supporting several key terrain data formats. In particular, the most important formats are Digital Terrain Elevation Data (DTED), which is the most commonly used format for elevation data, as well as “shape files”, which are utilized for the terrain features such as buildings, roads, rivers, lakes, and canopies. DTED and shape files are automatically translated into the internal LG-PACKAGE representation of the LG Abstract Board [3], [5]-[10]. An ability to automatically import such raw terrain information supports directly complex terrain models, terrain analysis, and MOUT operations (Section III). This ability provides a straightforward procedure for supplying terrain details for creating scenarios and domains that can take advantage of those details.

Quick Terrain Editing. For best application of the LG technology, LG-PACKAGE requires realistic terrain information containing elevation data as well as other terrain features such as buildings, roads, and rivers. Sometimes, such data is difficult to acquire for regions of interest and, in other cases, this data is outdated or of lower quality. While professional tools exist to build and update such databases, they are extremely expensive and can be difficult to use. LG-PACKAGE includes a component for editing terrain features to allow for quick modifications or construction of terrain databases directly from the GMI. This gives the GMI users an ability to perform calculations to quickly adjust the terrain source data in case of discrepancies with the real terrain they have noticed.

V. Sophisticated Simulations

Command Hierarchy. LG-PACKAGE allows the user to define aggregation of entities into the higher level virtual entities, the LG pieces [5], [6], as part of a command hierarchy. For example, individual tank entities of a platoon can be aggregated into the platoon entity (or unit), several of which can in turn be aggregated into the company entities. The GUI allows the user to visualize the current situation at any level of aggregation. In presence of the entities of various levels, the overall strategy/COA calculations are always performed by LG at the best level of resolution available in the hypergame, as defined by the user. This is especially useful if a multi-resolution LG hypergame is utilized because it permits to understand and assess the difference of decision making between high-level plans generated for the aggregated units, e.g., platoons, based on a low resolution map and detailed strategies generated for the finer-grain units or entities on a high resolution map. This can also be used to improve efficiency of calculations by simulating aggregated platoons when high resolution is not needed, and switching to individual entity representation during critical segments of the simulation. LG-PACKAGE allows the users to create teams, coalitions and introduce various types of collaboration within the LG hypergame.

Complex Pieces and Engagements. LG-PACKAGE supports a variety of simulation scenarios. For example, attrition and strength based scenarios are supported in addition to the standard Pk (Probability of Kill) based scenarios. This allows the user to define simulation where a single virtual entity, an LG piece, represents a group of real-world physical entities by specifying the strength (and/or size) of an entity. During an engagement the strength of such piece is decremented via an attrition calculation based on the combat effectiveness of the attack unit against the target unit. When the strength of a piece drops below a user specified threshold, the entity is considered destroyed. Another class of engagements requires modeling of decreased accuracy of weapons at greater distances. LG-PACKAGE supports the user definable “probability of hit” curves for each weapon that simulate decreased accuracy at longer ranges. Other parameters allow the user modifying values of probability of kill based on the effect of suppression due to hostile fire.

Batch Mode. LG-PACKAGE supports execution of simulation scenarios in a batch mode. The user can specify several initial positions and missions of the forces as well as the number of times to run each scenario. LG-PACKAGE executes each scenario the desired number of times and outputs detailed logs for each run as well as aggregated statistics. The optional logging features allow the user to request logging of nearly every type of event in a simulation including movements, engagements, sensor contacts, and communication exchanges.

Server Based Operation. LG-PACKAGE includes a capability to be executed in a server-client deployment in addition to the standard graphical standalone application. The LG-PACKAGE components can be executed as services on a server and accessed remotely from GMI over the network. This allows for a single LG-PACKAGE installation to be accessed by multiple simultaneous users. The server side components provide file storage, user access control, queue of request and interconnections to other 3rd party systems, such as a number of deployed military systems. A configuration utility is included to simplify configuration and maintenance of the server side components.

Help System. LG-PACKAGE contains a built-in comprehensive help system. This help system can be accessed from within any of the LG-PACKAGE GUI-enabled applications, such as GDK, GST, GRT, and GMI or it can be accessed independently. The content includes instructions for operating GUI, explanations for options available to the user of each of the software components, as well as tutorials and step-by-step instructions for performing most common user operations. The help system is continuously expanded to include more information as new features are introduced into software and by request from users for more information on specific topics. This help system is about to become context-sensitive.

Advanced GUI. The feedback from engineers and military experts after utilizing earlier versions of LG-PACKAGE allowed STILMAN to significantly improve the GUI. The current GUI permits to streamline user experience and provide additional visualization and editing tools. Such improvements include an ability to overlay any images over the 2D map display, draw freehand on the 2D map display, and measure distances. All the major editors enabling LG-PACKAGE GUI, including Mission Editor, Group Manager (Communications), Table of Organization (Command Hierarchy), and Piece Properties, are based on a unified hierarchical data presentation model and are highly transparent for the user. Further extensive collaboration with military users and SMEs allowed us to develop GMI, a light, highly mobile, streamlined interface that provides the most convenient, fast, and operationally correct method to manipulate all the components of LG-PACKAGE (Section I).

VI. Real Time Response

A number of defense systems require generating long term LG-based predictions in real time or near real time. These systems include RAID (Real-time Adversarial Intelligence and Decision-making) [1], [2], [3], [8], [9], FBCB2 (Force XXI Battle Command Brigade and Below) deployed on all the US Army Assault Vehicles, CPOF (Command Post of the Future) deployed at the top echelons of the US Army Battle Command, and Striker Embedded Training System. Additionally, various constructive simulation systems, such as OneSAF, require real time execution of the LG tools if those are utilized as an intelligent driver for both Blue and Red forces. Real time performance will be a must for applying LG for intelligent control of unmanned vehicles, aerial, water and ground. Those requirements were considered for developing LTP, the major optimization procedure.

Long Term Plans (LTP). LG-PACKAGE allows the user to calculate LTP, which are "deep" plans (estimates) including tightly interconnected estimates of the hostile COA and recommendations of the friendly COA. The standard operation of LG-PACKAGE is concerned with computing the most efficient action to be done by friendly and hostile forces at any given moment, and then repeat this computation cycle after every concurrent game move. This repetition leads to regenerating all the LG constructs, the LG zones and trajectories, and each time advancing the planning "distance horizon" over the abstract board [5]-[10]. These zones and trajectories serve as "rail tracks" for movement and actions of the LG pieces. The LTP procedure adds an ability to extend this technology by advancing the board horizon (and the respective time horizon) much further during one computation cycle without a significant increase of computation time.

Instead of regenerating all the LG constructs at each planning game move, the LTP procedure reuses trajectories and zones and drives pieces along these rail tracks until the first branching, i.e., until the moment when the first choice has to be made. After the initial zones and trajectories are generated, a path is chosen for every piece just as in the standard operation. However, after making a single move for each piece along its chosen path, as long as no branching point has been reached, new paths for entities do not have to be generated. Any piece that is still moving along its initially chosen path can continue movement without any new computations needed. On the other hand, when a critical event occurs – such as an engagement, discovery of new enemy forces, or a mission change – new zones and trajectories are generated for affected pieces and new paths are chosen. LG keeps track of both directly affected pieces, e.g., those involved in the engagement, as well as indirectly affected pieces, e.g., entities that are performing a collaborative task with the directly affected pieces. This allows for minimization of the set of pieces that require new trajectories and zones while still ensuring that this set includes all the pieces that may have to branch from the current path, i.e., may need to change their behavior. Analogously to the serial ABG [5], such reuse, is called "LG Zone Translation".

This optimization reuse permits to dramatically reduce the multi-move computation cycle down to 1-3 min while a standard one-move computation cycle requires 0.5-1 min. Specifically, the LTP procedure permits computing the likely course of events over a much longer period of time, the "time horizon", e.g., 250(!) game moves ahead, which may reflect several hours or days of astronomic time depending on the size of time interval for one move.

LTP contains all the required information about the "future". It includes initial positions of all the friendly and hostile pieces, as well as all the gradual changes, their estimated movements and actions, over the entire desired time horizon.

While LTP is meant to provide a deep look ahead into the future, even with all the predictive power of LG, that could include large number of branches (based on the outcome of engagements – random events, decisions made by the enemy, new sensor contacts, etc.), only one such branch of events is provided in each LTP. However, multiple LTPs can be computed based on slightly different input parameters to gain a broader understanding of the expected future up to the desired time horizon. LG-PACKAGE GUI (Fig. 1) provides an ability to view such estimated COA in the animated mode to help the user get an intuitive understanding of how the future is likely to unfold. Numerous experiments and analysis by SME (Subject Matter Experts) have shown that all the generated LTP are of high quality comparable or even better than those produced by the experienced experts, [2] and [3].

The main ideas and key algorithms that led to development of LTP have been thoroughly tested within DARPA RAID and US Army SBIR Phase II projects [2], [3], [8], [9].

VII. Utilizing LG-PACKAGE

The first organization that licensed the first release of LG-PACKAGE in 2004 was Dstl (Defence Science and Technology Lab) of the Ministry of Defence of UK. Subsequently, several versions of LG-PACKAGE were licensed to BAE Systems (UK) and Boeing (USA). A number of departments at Boeing including Boeing Integration Centers (BIC East and BIC West) utilized LG-PACKAGE. Various versions of customized LG-PACKAGE were licensed to the US DoD (Department of Defense) agencies including DARPA (Defense Advanced Research Projects Agency), JFCOM (Joint Forces Command) and NSWC (Naval Surface Warfare Center). Currently, the most active users of the latest versions of LG-PACKAGE are the three US Army organizations, DCGS-A (Distributed Common Ground System – Army), FBCB2 (Force XXI Battle Command Brigade and Below) and ARL (Army Research Lab for SIPRNET). Internationally, the key organization utilizing currently a universal version of LG-PACKAGE is SELEX Galileo, (UK), a Finmeccanica Company.

References

[1] DARPA RAID program, IPTO, 2004-08, www.darpa.mil/ipto/programs/raid/raid.asp.

[2] A. Kott, “Raiding The Enemy’s Mind,” *Military Information Technology*, Online Edition, Dec 29, 2007, www.military-information-technology.com/article.cfm?DocID=2287.

[3] *Linguistic Geometry Tools: LG-PACKAGE*, with 8 movies on Demo DVD, 60 pp., STILMAN, 2010. See also www.stilman-strategies.com.

[4] B. Stilman, “Linguistic Geometry of Complex Systems,” in *Abstracts of the 2nd Int. Symp. on Artificial Intelligence and Mathematics*, Ft. Lauderdale, FL, USA, Jan. 1992.

[5] B. Stilman, *Linguistic Geometry: From Search to Construction*, Kluwer Academic Publishers (now Springer-Verlag), 2000, 416 p.

[6] B. Stilman, V. Yakhnis, and O. Umanskiy, “Winning Strategies for Robotic Wars: Defense Applications of Linguistic Geometry,” *Artificial Life and Robotics*, Vol. 4, No. 3, pp. 148-155, 2000.

[7] B. Stilman, V. Yakhnis, and O. Umanskiy, “Knowledge Acquisition and Strategy Generation with LG Wargaming Tools,” *Int. J. of Comp. Intelligence and Appl.*, Vol. 2, No. 4, pp. 385-410, 2002.

[8] B. Stilman, V. Yakhnis, and O. Umanskiy, “Strategies in Large Scale Problems,” in *Adversarial Reasoning: Computational Approaches to Reading the Opponent's Mind*, Ed. by A. Kott and W. McEneaney, Chapter 3.3, Chapman & Hall/CRC, 2007, pp. 251-285.

[9] B. Stilman, V. Yakhnis, and O. Umanskiy, “Linguistic Geometry: The Age of Maturity”, *J. of Advanced Computational Intelligence and Intelligent Informatics,* Vol. 14, No. 6, 2010 (to be published).

[10] B. Stilman, V. Yakhnis, and O. Umanskiy, “Discovering Role of Linguistic Geometry,” in *Proc. of MICAI’2010*, Nov. 8-12, 2010, Pachuca, Mexico.

Summary Evaluation with and without References

Juan-Manuel Torres-Moreno, Horacio Saggion, Iria da Cunha, Eric SanJuan, and Patricia Velázquez-Morales

***Abstract*—We study a new content-based method for the evaluation of text summarization systems without human models which is used to produce system rankings. The research is carried out using a new content-based evaluation framework called FRESA to compute a variety of divergences among probability distributions. We apply our comparison framework to various well-established content-based evaluation measures in text summarization such as COVERAGE, RESPONSIVENESS, PYRAMIDS and ROUGE studying their associations in various text summarization tasks including generic multi-document summarization in English and French, focus-based multi-document summarization in English and generic single-document summarization in French and Spanish.**

***Index Terms*—Text summarization evaluation, content-based evaluation measures, divergences.**

I. INTRODUCTION

TEXT summarization evaluation has always been a complex and controversial issue in computational linguistics. In the last decade, significant advances have been made in this field as well as various evaluation measures have been designed. Two evaluation campaigns have been led by the U.S. agence DARPA. The first one, SUMMAC, ran from 1996 to 1998 under the auspices of the Tipster program [1], and the second one, entitled DUC (Document Understanding Conference) [2], was the main evaluation forum from 2000 until 2007. Nowadays, the Text Analysis Conference (TAC) [3] provides a forum for assessment of different information access technologies including text summarization.

Evaluation in text summarization can be extrinsic or intrinsic [4]. In an extrinsic evaluation, the summaries are assessed in the context of an specific task carried out by a human or a machine. In an intrinsic evaluation, the summaries are evaluated in reference to some ideal model. SUMMAC was mainly extrinsic while DUC and TAC followed an intrinsic evaluation paradigm. In an intrinsic evaluation, an automatically generated summary (*peer*) has to be compared with one or more reference summaries (*models*). DUC used an interface called SEE to allow human judges to compare a *peer* with a *model*. Thus, judges give a COVERAGE score to each *peer* produced by a system and the final system COVERAGE score is the average of the COVERAGE's scores asigned. These system's COVERAGE scores can then be used to rank summarization systems. In the case of query-focused summarization (e.g. when the summary should answer a question or series of questions) a RESPONSIVENESS score is also assigned to each summary, which indicates how responsive the summary is to the question(s).

Because manual comparison of peer summaries with model summaries is an arduous and costly process, a body of research has been produced in the last decade on automatic content-based evaluation procedures. Early studies used text similarity measures such as cosine similarity (with or without weighting schema) to compare peer and model summaries [5]. Various vocabulary overlap measures such as n-grams overlap or longest common subsequence between peer and model have also been proposed [6], [7]. The BLEU machine translation evaluation measure [8] has also been tested in summarization [9]. The DUC conferences adopted the ROUGE package for content-based evaluation [10]. ROUGE implements a series of recall measures based on n-gram co-occurrence between a peer summary and a set of model summaries. These measures are used to produce systems' rank. It has been shown that system rankings, produced by some ROUGE measures (e.g., ROUGE-2, which uses 2-grams), have a correlation with rankings produced using COVERAGE.

In recent years the PYRAMIDS evaluation method [11] has been introduced. It is based on the distribution of "content" of a set of model summaries. Summary Content Units (SCUs) are first identified in the model summaries, then each SCU receives a weight which is the number of models containing or expressing the same unit. Peer SCUs are identified in the peer, matched against model SCUs, and weighted accordingly. The PYRAMIDS score given to a peer is the ratio of the sum of the weights of its units and the sum of the weights of the best possible ideal summary with the same number of SCUs as the peer. The PYRAMIDS scores can be also used for ranking summarization systems. [11] showed that PYRAMIDS scores produced reliable system rankings when multiple (4 or more) models were used and that PYRAMIDS rankings correlate with rankings produced by ROUGE-2 and ROUGE-SU2 (i.e. ROUGE with skip 2-grams). However, this method requires the creation

Manuscript received June 8, 2010. Manuscript accepted for publication July 25, 2010.

Juan-Manuel Torres-Moreno is with LIA/Université d'Avignon, France and École Polytechnique de Montréal, Canada (juan-manuel.torres@univ-avignon.fr).

Eric SanJuan is with LIA/Université d'Avignon, France (eric.sanjuan@univ-avignon.fr).

Horacio Saggion is with DTIC/Universitat Pompeu Fabra, Spain (horacio.saggion@upf.edu).

Iria da Cunha is with IULA/Universitat Pompeu Fabra, Spain; LIA/Université d'Avignon, France and Instituto de Ingeniería/UNAM, Mexico (iria.dacunha@upf.edu).

Patricia Velázquez-Morales is with VM Labs, France (patricia_velazquez@yahoo.com).

of models and the identification, matching, and weighting of SCUs in both: models and peers.

[12] evaluated the effectiveness of the Jensen-Shannon ($\mathcal{JS}$) [13] theoretic measure in predicting systems ranks in two summarization tasks: query-focused and update summarization. They have shown that ranks produced by PYRAMIDS and those produced by $\mathcal{JS}$ measure correlate. However, they did not investigate the effect of the measure in summarization tasks such as generic multi-document summarization (DUC 2004 Task 2), biographical summarization (DUC 2004 Task 5), opinion summarization (TAC 2008 OS), and summarization in languages other than English.

In this paper we present a series of experiments aimed at a better understanding of the value of the $\mathcal{JS}$ divergence for ranking summarization systems. We have carried out experimentation with the proposed measure and we have verified that in certain tasks (such as those studied by [12]) there is a strong correlation among PYRAMIDS, RESPONSIVENESS and the $\mathcal{JS}$ divergence, but as we will show in this paper, there are datasets in which the correlation is not so strong. We also present experiments in Spanish and French showing positive correlation between the $\mathcal{JS}$ and ROUGE which is the *de facto* evaluation measure used in evaluation of non-English summarization. To the best of our knowledge this is the more extensive set of experiments interpreting the value of evaluation without human models.

The rest of the paper is organized in the following way: First in Section II we introduce related work in the area of content-based evaluation identifying the departing point for our inquiry; then in Section III we explain the methodology adopted in our work and the tools and resources used for experimentation. In Section IV we present the experiments carried out together with the results. Section V discusses the results and Section VI concludes the paper and identifies future work.

II. Related Work

One of the first works to use content-based measures in text summarization evaluation is due to [5], who presented an evaluation framework to compare rankings of summarization systems produced by recall and cosine-based measures. They showed that there was weak correlation among rankings produced by recall, but that content-based measures produce rankings which were strongly correlated. This put forward the idea of using directly the full document for comparison purposes in text summarization evaluation. [6] presented a set of evaluation measures based on the notion of vocabulary overlap including n-gram overlap, cosine similarity, and longest common subsequence, and they applied them to multi-document summarization in English and Chinese. However, they did not evaluate the performance of the measures in different summarization tasks. [7] also compared various evaluation measures based on vocabulary overlap. Although these measures were able to separate random from non-random systems, no clear conclusion was reached on the value of each of the studied measures.

Nowadays, a widespread summarization evaluation framework is ROUGE [14], which offers a set of statistics that compare peer summaries with models. It counts co-occurrences of n-grams in peer and models to derive a score. There are several statistics depending on the used n-grams and the text processing applied to the input texts (e.g., lemmatization, stop-word removal).

[15] proposed a method of evaluation based on the use of "distances" or divergences between two probability distributions (the distribution of units in the automatic summary and the distribution of units in the model summary). They studied two different Information Theoretic measures of divergence: the Kullback-Leibler ($\mathcal{KL}$) [16] and Jensen-Shannon ($\mathcal{JS}$) [13] divergences. $\mathcal{KL}$ computes the divergence between probability distributions P and Q in the following way:

$$D_{\mathcal{KL}}(P||Q) = \frac{1}{2}\sum_{w} P_w \log_2 \frac{P_w}{Q_w} \tag{1}$$

While $\mathcal{JS}$ divergence is defined as follows:

$$D_{\mathcal{JS}}(P||Q) = \frac{1}{2}\sum_{w} P_w \log_2 \frac{2P_w}{P_w+Q_w} + Q_w \log_2 \frac{2Q_w}{P_w+Q_w} \tag{2}$$

These measures can be applied to the distribution of units in system summaries P and reference summaries Q. The value obtained may be used as a score for the system summary. The method has been tested by [15] over the DUC 2002 corpus for single and multi-document summarization tasks showing good correlation among divergence measures and both coverage and ROUGE rankings.

[12] went even further and, as in [5], they proposed to compare directly the distribution of words in full documents with the distribution of words in automatic summaries to derive a content-based evaluation measure. They found a high correlation between rankings produced using models and rankings produced without models. This last work is the departing point for our inquiry into the value of measures that do not rely on human models.

III. Methodology

The followed methodology in this paper mirrors the one adopted in past work (e.g. [5], [7], [12]). Given a particular summarization task T, p data points to be summarized with input material $\{I_i\}_{i=0}^{p-1}$ (e.g. document(s), question(s), topic(s)), s peer summaries $\{\text{SUM}_{i,k}\}_{k=0}^{s-1}$ for input i, and m model summaries $\{\text{MODEL}_{i,j}\}_{j=0}^{m-1}$ for input i, we will compare rankings of the s peer summaries produced by various evaluation measures. Some measures that we use compare summaries with n of the m models:

$$\text{MEASURE}_M(\text{SUM}_{i,k}, \{\text{MODEL}_{i,j}\}_{j=0}^{n-1}) \tag{3}$$

while other measures compare peers with all or some of the input material:

$$\text{MEASURE}_M(\text{SUM}_{i,k}, I'_i) \qquad (4)$$

where I'_i is some subset of input I_i. The values produced by the measures for each summary $\text{SUM}_{i,k}$ are averaged for each system $k = 0, \ldots, s-1$ and these averages are used to produce a ranking. Rankings are then compared using Spearman Rank correlation [17] which is used to measure the degree of association between two variables whose values are used to rank objects. We have chosen to use this correlation to compare directly results to those presented in [12]. Computation of correlations is done using the *Statistics-RankCorrelation-0.12* package[1], which computes the rank correlation between two vectors. We also verified the good conformity of the results with the correlation test of Kendall τ calculated with the statistical software R. The two nonparametric tests of Spearman and Kendall do not really stand out as the treatment of ex-æquo. The good correspondence between the two tests shows that they do not introduce bias in our analysis. Subsequently will mention only the ρ of Sperman more widely used in this field.

A. Tools

We carry out experimentation using a new summarization evaluation framework: FRESA –FRamework for Evaluating Summaries Automatically–, which includes document-based summary evaluation measures based on probabilities distribution[2]. As in the ROUGE package, FRESA supports different n-grams and skip n-grams probability distributions. The FRESA environment can be used in the evaluation of summaries in English, French, Spanish and Catalan, and it integrates filtering and lemmatization in the treatment of summaries and documents. It is developed in Perl and will be made publicly available. We also use the ROUGE package [10] to compute various ROUGE statistics in new datasets.

B. Summarization Tasks and Data Sets

We have conducted our experimentation with the following summarization tasks and data sets:

1) Generic multi-document-summarization in English (production of a short summary of a cluster of related documents) using data from DUC'04[3], task 2: 50 clusters, 10 documents each – 294,636 words.
2) Focused-based summarization in English (production of a short focused multi-document summary focused on the question "who is X?", where X is a person's name) using data from the DUC'04 task 5: 50 clusters, 10 documents each plus a target person name – 284,440 words.
3) Update-summarization task that consists of creating a summary out of a cluster of documents and a topic. Two sub-tasks are considered here: A) an initial summary has to be produced based on an initial set of documents and topic; B) an update summary has to be produced from a different (but related) cluster assuming documents used in A) are known. The English TAC'08 Update Summarization dataset is used, which consists of 48 topics with 20 documents each – 36,911 words.
4) Opinion summarization where systems have to analyze a set of blog articles and summarize the opinions about a target in the articles. The TAC'08 Opinion Summarization in English[4] data set (taken from the Blogs06 Text Collection) is used: 25 clusters and targets (i.e., target entity and questions) were used – 1,167,735 words.
5) Generic single-document summarization in Spanish using the *Medicina Clínica*[5] corpus, which is composed of 50 medical articles in Spanish, each one with its corresponding author abstract – 124,929 words.
6) Generic single document summarization in French using the "Canadien French Sociological Articles" corpus from the journal *Perspectives interdisciplinaires sur le travail et la santé* (PISTES)[6]. It contains 50 sociological articles in French, each one with its corresponding author abstract – 381,039 words.
7) Generic multi-document-summarization in French using data from the RPM2[7] corpus [18], 20 different themes consisting of 10 articles and 4 abstracts by reference thematic – 185,223 words.

For experimentation in the TAC and the DUC datasets we use directly the peer summaries produced by systems participating in the evaluations. For experimentation in Spanish and French (single and multi-document summarization) we have created summaries at a similar ratio to those of reference using the following systems:

– *ENERTEX* [19], a summarizer based on a theory of textual energy;
– *CORTEX* [20], a single-document sentence extraction system for Spanish and French that combines various statistical measures of relevance (angle between sentence and topic, various Hamming weights for sentences, etc.) and applies an optimal decision algorithm for sentence selection;
– *SUMMTERM* [21], a terminology-based summarizer that is used for summarization of medical articles and uses specialized terminology for scoring and ranking sentences;
– *REG* [22], summarization system based on an greedy algorithm;

[1] http://search.cpan.org/~gene/Statistics-RankCorrelation-0.12/

[2] FRESA is available at: http://lia.univavignon.fr/fileadmin/axes/TALNE/Ressources.html

[3] http://www-nlpir.nist.gov/projects/duc/guidelines/2004.html

[4] http://www.nist.gov/tac/data/index.html

[5] http://www.elsevier.es/revistas/ctl_servlet?_f=7032&revistaid=2

[6] http://www.pistes.uqam.ca/

[7] http://www-labs.sinequa.com/rpm2

- $\mathcal{JS}$ summarizer, a summarization system that scores and ranks sentences according to their Jensen-Shannon divergence to the source document;
- a *lead-based* summarization system that selects the lead sentences of the document;
- a *random-based* summarization system that selects sentences at random;
- *Open Text Summarizer* [23], a multi-lingual summarizer based on the frequency and
- commercial systems: *Word*, *SSSummarizer*[8], *Pertinence*[9] and *Copernic*[10].

C. Evaluation Measures

The following measures derived from human assessment of the content of the summaries are used in our experiments:

- COVERAGE is understood as the degree to which one peer summary conveys the same information as a model summary [2]. COVERAGE was used in DUC evaluations. This measure is used as indicated in equation 3 using human references or models.
- RESPONSIVENESS ranks summaries in a 5-point scale indicating how well the summary satisfied a given information need [2]. It is used in focused-based summarization tasks. This measure is used as indicated in equation 4 since a human judges the summary with respect to a given input "user need" (e.g., a question). RESPONSIVENESS was used in DUC and TAC evaluations.
- PYRAMIDS [11] is a content assessment measure which compares content units in a peer summary to weighted content units in a set of model summaries. This measure is used as indicated in equation 3 using human references or models. PYRAMIDS is the adopted metric for content-based evaluation in the TAC evaluations.

For DUC and TAC datasets the values of these measures are available and we used them directly. We used the following automatic evaluation measures in our experiments:

- ROUGE [14], which is a recall metric that takes into account n-grams as units of content for comparing peer and model summaries. The ROUGE formula specified in [10] is as follows:

$$\text{ROUGE-}n(\text{R}, M) = \frac{\sum_m \in M \sum_{n-\text{gram} \in P} \text{count}_{\text{match}}(n-\text{gram})}{\sum_m \in M \sum \text{count(n-gram)}} \quad (5)$$

  where R is the summary to be evaluated, M is the set of model (human) summaries, $\text{count}_{\text{match}}$ is the number of common n-grams in m and P, and count is the number of n-grams in the model summaries. For the experiments presented here we used uni-grams, 2-grams, and the skip 2-grams with maximum skip distance of 4 (ROUGE-1, ROUGE-2 and ROUGE-SU4). ROUGE is used to compare a peer summary to a set of model summaries in our framework (as indicated in equation 3).
- Jensen-Shannon divergence formula given in Equation 2 is implemented in our FRESA package with the following specification (Equation 6) for the probability distribution of words w.

$$P_w = \frac{C_w^T}{N}$$

$$Q_w = \begin{cases} \frac{C_w^S}{N_S} & \text{if } w \in S \\ \frac{C_w^T + \delta}{N + \delta * B} & \text{otherwise} \end{cases} \quad (6)$$

Where P is the probability distribution of words w in text T and Q is the probability distribution of words w in summary S; N is the number of words in text and summary $N = N_T + N_S$, $B = 1.5|V|$, C_w^T is the number of words in the text and C_w^S is the number of words in the summary. For smoothing the summary's probabilities we have used $\delta = 0.005$. We have also implemented other smoothing approaches (e.g. Good-Turing [24], that uses the CPAN Perl's Statistics-Smoothing-SGT-2.1.2 package[11]) in FRESA, but we do not use them in the experiments reported here. Following the ROUGE approach, in addition to word uni-grams we use 2-grams and skip n-grams computing divergences such as $\mathcal{JS}$ (using uni-grams) $\mathcal{JS}_2$ (using 2-grams), $\mathcal{JS}_4$ (using the skip n-grams of ROUGE-SU4), and $\mathcal{JS}_M$ which is an average of the $\mathcal{JS}_i$. $\mathcal{JS}$s measures are used to compare a peer summary to its source document(s) in our framework (as indicated in equation 4). In the case of summarization of multiple documents, these are concatenated (in the given input order) to form a single input from which probabilities are computed.

IV. EXPERIMENTS AND RESULTS

We first replicated the experiments presented in [12] to verify that our implementation of $\mathcal{JS}$ produced correlation results compatible with that work. We used the TAC'08 Update Summarization data set and computed $\mathcal{JS}$ and ROUGE measures for each peer summary. We produced two system rankings (one for each measure), which were compared to rankings produced using the manual PYRAMIDS and RESPONSIVENESS scores. Spearman correlations were computed among the different rankings. The results are presented in Table I. These results confirm a high correlation among PYRAMIDS, RESPONSIVENESS and $\mathcal{JS}$. We also verified high correlation between $\mathcal{JS}$ and ROUGE-2 (0.83 Spearman correlation, not shown in the table) in this task and dataset.

Then, we experimented with data from DUC'04, TAC'08 Opinion Summarization pilot task as well as single and

[8] http://www.kryltech.com/summarizer.htm

[9] http://www.pertinence.net

[10] http://www.copernic.com/en/products/summarizer

[11] http://search.cpan.org/~bjoernw/Statistics-Smoothing-SGT-2.1.2/

TABLE I
Spearman correlation of content-based measures in TAC'08 Update Summarization task

Mesure	Pyramids	p-value	Responsiveness	p-value
Rouge-2	0.96	$p < 0.005$	0.92	$p < 0.005$
$\mathcal{JS}$	0.85	$p < 0.005$	0.74	$p < 0.005$

multi-document summarization in Spanish and French. In spite of the fact that the experiments for French and Spanish corpora use less data points (i.e., less summarizers per task) than for English, results are still quite significant. For DUC'04, we computed the $\mathcal{JS}$ measure for each peer summary in tasks 2 and 5 and we used $\mathcal{JS}$, Rouge, Coverage and Responsiveness scores to produce systems' rankings. The various Spearman's rank correlation values for DUC'04 are presented in Tables II (for task 2) and III (for task 5). For task 2, we have verified a strong correlation between $\mathcal{JS}$ and Coverage. For task 5, the correlation between $\mathcal{JS}$ and Coverage is weak, and that between $\mathcal{JS}$ and Responsiveness is weak and negative.

Although the Opinion Summarization (OS) task is a new type of summarization task and its evaluation is a complicated issue, we have decided to compare $\mathcal{JS}$ rankings with those obtained using Pyramids and Responsiveness in TAC'08. Spearman's correlation values are listed in Table IV. As it can be seen, there is weak and negative correlation of $\mathcal{JS}$ with both Pyramids and Responsiveness. Correlation between Pyramids and Responsiveness rankings is high for this task (0.71 Spearman's correlation value).

For experimentation in mono-document summarization in Spanish and French, we have run 11 multi-lingual summarization systems; for experimentation in French, we have run 12 systems. In both cases, we have produced summaries at a compression rate close to the compression rate of the authors' provided abstracts. We have then computed $\mathcal{JS}$ and Rouge measures for each summary and we have averaged the measure's values for each system. These averages were used to produce rankings per each measure. We computed Spearman's correlations for all pairs of rankings.

Results are presented in Tables V, VI and VII. All results show medium to strong correlation between the $\mathcal{JS}$ measures and Rouge measures. However the $\mathcal{JS}$ measure based on uni-grams has lower correlation than $\mathcal{JS}$s which use n-grams of higher order. Note that table VII presents results for generic multi-document summarization in French, in this case correlation scores are lower than correlation scores for single-document summarization in French, a result which may be expected given the diversity of input in multi-document summarization.

V. Discussion

The departing point for our inquiry into text summarization evaluation has been recent work on the use of content-based evaluation metrics that do not rely on human models but that compare summary content to input content directly [12]. We have some positive and some negative results regarding the direct use of the full document in content-based evaluation.

We have verified that in both generic muti-document summarization and in topic-based multi-document summarization in English correlation among measures that use human models (Pyramids, Responsiveness and Rouge) and a measure that does not use models ($\mathcal{JS}$ divergence) is strong. We have found that correlation among the same measures is weak for summarization of biographical information and summarization of opinions in blogs. We believe that in these cases content-based measures should be considered, in addition to the input document, the summarization task (i.e. text-based representation, description) to better assess the content of the peers [25], the task being a determinant factor in the selection of content for the summary.

Our multi-lingual experiments in generic single-document summarization confirm a strong correlation among the $\mathcal{JS}$ divergence and Rouge measures. It is worth noting that Rouge is in general the chosen framework for presenting content-based evaluation results in non-English summarization.

For the experiments in Spanish, we are conscious that we only have one model summary to compare with the peers. Nevertheless, these models are the corresponding abstracts written by the authors. As the experiments in [26] show, the professionals of a specialized domain (as, for example, the medical domain) adopt similar strategies to summarize their texts and they tend to choose roughly the same content chunks for their summaries. Previous studies have shown that author abstracts are able to reformulate content with fidelity [27] and these abstracts are ideal candidates for comparison purposes. Because of this, the summary of the author of a medical article can be taken as reference for summaries evaluation. It is worth noting that there is still debate on the number of models to be used in summarization evaluation [28]. In the French corpus PISTES, we suspect the situation is similar to the Spanish case.

VI. Conclusions and Future Work

This paper has presented a series of experiments in content-based measures that do not rely on the use of model summaries for comparison purposes. We have carried out extensive experimentation with different summarization tasks drawing a clearer picture of tasks where the measures could be applied. This paper makes the following contributions:

- We have shown that if we are only interested in ranking summarization systems according to the content of their automatic summaries, there are tasks were models could be subtituted by the full document in the computation of the $\mathcal{JS}$ measure obtaining reliable rankings. However, we have also found that the substitution of models by full-documents is not always advisable. We have

TABLE II
SPEARMAN ρ OF CONTENT-BASED MEASURES WITH COVERAGE IN DUC'04 TASK 2

Mesure	COVERAGE	p-value
ROUGE-2	0.79	$p < 0.0050$
$\mathcal{JS}$	0.68	$p < 0.0025$

TABLE III
SPEARMAN ρ OF CONTENT-BASED MEASURES IN DUC'04 TASK 5

Mesure	COVERAGE	p-value	RESPONSIVENESS	p-value
ROUGE-2	0.78	$p < 0.001$	0.44	$p < 0.05$
$\mathcal{JS}$	0.40	$p < 0.050$	-0.18	$p < 0.25$

TABLE IV
SPEARMAN ρ OF CONTENT-BASED MEASURES IN TAC'08 OS TASK

Mesure	PYRAMIDS	p-value	RESPONSIVENESS	p-value
$\mathcal{JS}$	-0.13	$p < 0.25$	-0.14	$p < 0.25$

TABLE V
SPEARMAN ρ OF CONTENT-BASED MEASURES WITH ROUGE IN THE *Medicina Clínica* CORPUS (SPANISH)

Mesure	ROUGE-1	p-value	ROUGE-2	p-value	ROUGE-SU4	p-value
$\mathcal{JS}$	0.56	$p < 0.100$	0.46	$p < 0.100$	0.45	$p < 0.200$
$\mathcal{JS}_2$	0.88	$p < 0.001$	0.80	$p < 0.002$	0.81	$p < 0.005$
$\mathcal{JS}_4$	0.88	$p < 0.001$	0.80	$p < 0.002$	0.81	$p < 0.005$
$\mathcal{JS}_M$	0.82	$p < 0.005$	0.71	$p < 0.020$	0.71	$p < 0.010$

found weak correlation among different rankings in complex summarization tasks such as the summarization of biographical information and the summarization of opinions.

– We have also carried out large-scale experiments in Spanish and French which show positive medium to strong correlation among system's ranks produced by ROUGE and divergence measures that do not use the model summaries.

– We have also presented a new framework, FRESA, for the computation of measures based on $\mathcal{JS}$ divergence. Following the ROUGE approach, FRESA package use word uni-grams, 2-grams and skip n-grams computing divergences. This framework will be available to the community for research purposes.

Although we have made a number of contributions, this paper leaves many open questions than need to be addressed. In order to verify correlation between ROUGE and $\mathcal{JS}$, in the short term we intend to extend our investigation to other languages such as Portuguese and Chinesse for which we have access to data and summarization technology. We also plan to apply FRESA to the rest of the DUC and TAC summarization tasks, by using several smoothing techniques. As a novel idea, we contemplate the possibility of adapting the evaluation framework for the phrase compression task [29], which, to our knowledge, does not have an efficient evaluation measure. The main idea is to calculate $\mathcal{JS}$ from an automatically-compressed sentence taking the complete sentence by reference. In the long term, we plan to incorporate a representation of the task/topic in the calculation of measures. To carry out these comparisons, however, we are dependent on the existence of references.

FRESA will also be used in the new question-answer task campaign INEX'2010 (http://www.inex.otago.ac.nz/tracks/qa/qa.asp) for the evaluation of long answers. This task aims to answer a question by extraction and agglomeration of sentences in Wikipedia. This kind of task corresponds to those for which we have found a high correlation among the measures $\mathcal{JS}$ and evaluation methods with human intervention. Moreover, the $\mathcal{JS}$ calculation will be among the summaries produced and a representative set of relevant passages from Wikipedia. FRESA will be used to compare three types of systems, although different tasks: the multi-document summarizer guided by a query, the search systems targeted information (focused IR) and the question answering systems.

ACKNOWLEDGMENT

We are grateful to the Programa Ramón y Cajal from Ministerio de Ciencia e Innovación, Spain. This work is partially supported by: a postdoctoral grant from the National Program for Mobility of Research Human Resources (National Plan of Scientific Research, Development and Innovation 2008-2011, Ministerio de Ciencia e Innovación, Spain); the research project CONACyT, number 82050, and the research project PAPIIT-DGAPA (Universidad Nacional Autónoma de México), number IN403108.

TABLE VI
Spearman ρ of content-based measures with Rouge in the PISTES Corpus (French)

Mesure	Rouge-1	p-value	Rouge-2	p-value	Rouge-SU4	p-value
$\mathcal{JS}$	0.70	$p < 0.050$	0.73	$p < 0.05$	0.73	$p < 0.500$
$\mathcal{JS}_2$	0.93	$p < 0.002$	0.86	$p < 0.01$	0.86	$p < 0.005$
$\mathcal{JS}_4$	0.83	$p < 0.020$	0.76	$p < 0.05$	0.76	$p < 0.050$
$\mathcal{JS}_M$	0.88	$p < 0.010$	0.83	$p < 0.02$	0.83	$p < 0.010$

TABLE VII
Spearman ρ of content-based measures with Rouge in the RPM2 Corpus (French)

Measure	Rouge-1	p-value	Rouge-2	p-value	Rouge-SU4	p-value
$\mathcal{JS}$	0.830	$p < 0.002$	0.660	$p < 0.05$	0.741	$p < 0.01$
$\mathcal{JS}_2$	0.800	$p < 0.005$	0.590	$p < 0.05$	0.680	$p < 0.02$
$\mathcal{JS}_4$	0.750	$p < 0.010$	0.520	$p < 0.10$	0.620	$p < 0.05$
$\mathcal{JS}_M$	0.850	$p < 0.002$	0.640	$p < 0.05$	0.740	$p < 0.01$

References

[1] I. Mani, G. Klein, D. House, L. Hirschman, T. Firmin, and B. Sundheim, "Summac: a text summarization evaluation," *Natural Language Engineering*, vol. 8, no. 1, pp. 43–68, 2002.

[2] P. Over, H. Dang, and D. Harman, "DUC in context," *IPM*, vol. 43, no. 6, pp. 1506–1520, 2007.

[3] *Proceedings of the Text Analysis Conference*. Gaithesburg, Maryland, USA: NIST, November 17-19 2008.

[4] K. Spärck Jones and J. Galliers, *Evaluating Natural Language Processing Systems, An Analysis and Review*, ser. Lecture Notes in Computer Science. Springer, 1996, vol. 1083.

[5] R. L. Donaway, K. W. Drummey, and L. A. Mather, "A comparison of rankings produced by summarization evaluation measures," in *NAACL Workshop on Automatic Summarization*, 2000, pp. 69–78.

[6] H. Saggion, D. Radev, S. Teufel, and W. Lam, "Meta-evaluation of Summaries in a Cross-lingual Environment using Content-based Metrics," in *COLING 2002*, Taipei, Taiwan, August 2002, pp. 849–855.

[7] D. R. Radev, S. Teufel, H. Saggion, W. Lam, J. Blitzer, H. Qi, A. Çelebi, D. Liu, and E. Drábek, "Evaluation challenges in large-scale document summarization," in *ACL'03*, 2003, pp. 375–382.

[8] K. Papineni, S. Roukos, T. Ward, , and W. J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *ACL'02*, 2002, pp. 311–318.

[9] K. Pastra and H. Saggion, "Colouring summaries BLEU," in *Evaluation Initiatives in Natural Language Processing*. Budapest, Hungary: EACL, 14 April 2003.

[10] C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," in *Text Summarization Branches Out: ACL-04 Workshop*, M.-F. Moens and S. Szpakowicz, Eds., Barcelona, July 2004, pp. 74–81.

[11] A. Nenkova and R. J. Passonneau, "Evaluating Content Selection in Summarization: The Pyramid Method," in *HLT-NAACL*, 2004, pp. 145–152.

[12] A. Louis and A. Nenkova, "Automatically Evaluating Content Selection in Summarization without Human Models," in *Empirical Methods in Natural Language Processing*, Singapore, August 2009, pp. 306–314. [Online]. Available: http://www.aclweb.org/anthology/D/D09/D09-1032

[13] J. Lin, "Divergence Measures based on the Shannon Entropy," *IEEE Transactions on Information Theory*, vol. 37, no. 145-151, 1991.

[14] C.-Y. Lin and E. Hovy, "Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics," in *HLT-NAACL*. Morristown, NJ, USA: Association for Computational Linguistics, 2003, pp. 71–78.

[15] C.-Y. Lin, G. Cao, J. Gao, and J.-Y. Nie, "An information-theoretic approach to automatic evaluation of summaries," in *HLT-NAACL*, Morristown, USA, 2006, pp. 463–470.

[16] S. Kullback and R. Leibler, "On information and sufficiency," *Ann. of Math. Stat.*, vol. 22, no. 1, pp. 79–86, 1951.

[17] S. Siegel and N. Castellan, *Nonparametric Statistics for the Behavioral Sciences*. McGraw-Hill, 1998.

[18] C. de Loupy, M. Guégan, C. Ayache, S. Seng, and J.-M. Torres-Moreno, "A French Human Reference Corpus for multi-documents summarization and sentence compression," in *LREC'10*, vol. 2, Malta, 2010, p. In press.

[19] S. Fernandez, E. SanJuan, and J.-M. Torres-Moreno, "Textual Energy of Associative Memories: performants applications of Enertex algorithm in text summarization and topic segmentation," in *MICAI'07*, 2007, pp. 861–871.

[20] J.-M. Torres-Moreno, P. Velázquez-Morales, and J.-G. Meunier, "Condensés de textes par des méthodes numériques," in *JADT'02*, vol. 2, St Malo, France, 2002, pp. 723–734.

[21] J. Vivaldi, I. da Cunha, J.-M. Torres-Moreno, and P. Velázquez-Morales, "Automatic summarization using terminological and semantic resources," in *LREC'10*, vol. 2, Malta, 2010, p. In press.

[22] J.-M. Torres-Moreno and J. Ramirez, "REG : un algorithme glouton appliqué au résumé automatique de texte," in *JADT'10*. Rome, 2010, p. In press.

[23] V. Yatsko and T. Vishnyakov, "A method for evaluating modern systems of automatic text summarization," *Automatic Documentation and Mathematical Linguistics*, vol. 41, no. 3, pp. 93–103, 2007.

[24] C. D. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*. Cambridge, Massachusetts: The MIT Press, 1999.

[25] K. Spärck Jones, "Automatic summarising: The state of the art," *IPM*, vol. 43, no. 6, pp. 1449–1481, 2007.

[26] I. da Cunha, L. Wanner, and M. T. Cabré, "Summarization of specialized discourse: The case of medical articles in spanish," *Terminology*, vol. 13, no. 2, pp. 249–286, 2007.

[27] C.-K. Chuah, "Types of lexical substitution in abstracting," in *ACL Student Research Workshop*. Toulouse, France: Association for Computational Linguistics, 9-11 July 2001 2001, pp. 49–54.

[28] K. Owkzarzak and H. T. Dang, "Evaluation of automatic summaries: Metrics under varying data conditions," in *UCNLG+Sum'09*, Suntec, Singapore, August 2009, pp. 23–30.

[29] K. Knight and D. Marcu, "Statistics-based summarization-step one: Sentence compression," in *Proceedings of the National Conference on Artificial Intelligence*. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2000, pp. 703–710.

The Role of Automation in Instruction

Joseph M. Scandura

***Abstract*—More and more things that humans used to do can be automated on computer. In each case, complex tasks have been automated – not to the extent that they can be done as well as humans, but better. I will draw and develop parallels to education – showing how and why advances in the Structural Learning Theory (SLT) and the AuthorIT development and TutorIT delivery technologies based thereon make it possible not only to duplicate many of the things that human math tutors can do but to do them better. Specifically, I will show how and why TutorIT can now do a better job than most if not all human tutors in providing more effective and efficient tutoring on essentially any well defined skill. I also will show why this approach has the potential to also match or exceed human tutoring on ill-defined learning in the future.**

***Index terms*—Automation, instruction, computer-aided learning.**

I. Introduction

Automation involves the use of control and information systems reducing the need for human intervention. According to Wikipedia, automation is a step beyond *mechanization*. Whereas mechanization provided human operators with machinery to assist them with the *muscular* requirements of work, *automation* greatly reduces the need for human sensory and *mental* requirements as well. AI, for example, was founded on the claim that a central property of human intelligence can be so precisely described that it can be simulated by a machine[1] Proponents have long claimed that increases in computational power will eventually overtake the human mind. IBM's Big Blue beating Chess masters is often sighted to support this claim. On the other hand, most AI research has become increasingly technical and specialized.

Progress is being made in subfields, where solutions to specific problems can be automated. This is a pattern that has been replicated in almost every software intensive application area. Who today would compute taxes using paper and pencil? Keep records on a rolodex? … Today, we have immediate access to almost any information in databases, instant communication throughout the world and the ability to quickly find information on almost any topic – at least if it occurred or was documented after the advent of the world wide web.

Intelligent behavior has not happened, however, except in a very special sense: More and more things that humans used to have to do themselves can be automated on computer. In each case, increasingly complex tasks have been automated – not to the extent that they can be done as well as humans, but better.

My goal today is to draw and develop parallels to education. Major attention is being given to immersive, often game-like environments. Students are placed in various problem solving situations – and allowed to either explore on their own or with various kinds of hints (today typically called "scaffolding"). The big questions here are what kinds of hints/scaffolding will be of (most) help and when should it be given?

Other tools such as Texas Instrument's TI-Nspire tackle the problem from the other end. Rather than hints, calculators serve as tools students can use to facilitate problem solving, serve as prerequisites – as more or less comprehensive foundational skills on which learners may build.

Scaffolding and prerequisites both play a central role in all learning systems. The main problem is that good tutoring systems difficult and expensive to build. Moreover, their educational benefits are difficult and expensive to evaluate. Determining effectiveness and efficiency invariably requires direct (and often expensive) empirical evaluation. The results are rarely if ever as good as what a human tutor can do, and comparisons with classroom instruction are often hard to evaluate.

Instructional design models help. Among other things they help identify what must be mastered for success and what can be assumed on entry. Computer Based Instruction (CBI) systems build on assumed prerequisites and are directed at what must be learned. After years of effort, beginning with Control Data's work (under the leadership of William Norris) in the early 1960s, the best CBI is limited to providing pretests to identify areas of weakness, providing instruction aimed at deficiencies and following up with post tests to determine how much has been learned.

ALEKS is one of the better commercially available CBI systems. In ALEKS and other advanced CBI systems (e.g., Paquette, 2007, 2009) to-be-acquired knowledge is represented in terms of relational models.

ITS research goes further, attempting to duplicate or model what a good tutor can do – by adjusting diagnosis and remediation dynamically during instruction. ITS focus on modeling and diagnosing what is going on in learner minds (e.g., Anderson, 1993; cf. Koedinger et al, 1997; Scandura et al, 2007). Assumptions are made both about what knowledge

Manuscript received August 1, 2010. Manuscript accepted for publication September 12, 2010.

The author is Director of Research, MERGE Research Institute and Emeritus Professor, University of Pennsylvania, USA (JosephScandura@comcast.net).

[1] This definition derives from John McCarthy's view of AI "the science and technology of making intelligent systems". Early researchers at Carnegie Mellon (Newell & Simon, 1972) tended to view AI more in terms of simulating human thought -- as trying to describe human cognition in precise terms similar to those required to program a computer, believing that doing so would help to reveal fundamental properties of human intelligence

might be in learner minds and learning mechanisms controlling the way those productions are used in producing behavior and learning.

Identifying the productions involved in any given domain is a difficult task. Specifying learning mechanisms is even harder. Recognizing these complexities Carnegie Learning credits Anderson's evolving ACT theories, but increasingly has focused on integrating ITS with print materials to make them educationally palatable (i.e., more closely aligned with what goes on in classrooms).

The difficulties do not stop there. Ohlsson noted as early as 1987 that specifying remedial actions – what to teach is much harder than modeling and diagnosis. As in CBI, pedagogical decisions in ITS necessarily depend on the subject matter being taught – on semantics of the content. Each content domain requires its own unique set of pedagogical decisions. It is not surprising in this context that Ohlsson and Mitrovic found common cause in developing Constraint Based Modeling (CBM, 2007). CBM is a simplified alternative to ITS based on production systems in which the focus is on constraints that must be met during the course of instruction – not on the cognitive constructs (productions) responsible (for meeting those constraints).

From their inceptions, the Holy Grail in CBI and ITS is to duplicate what good teachers do. As shown by Bloom (1984) the best human tutors can improve mastery in comparison to normal instruction by 2 sigmas. This goal has been broadly influential but never achieved through automation. The limited success of CBI, combined with the complexities and cost inefficiencies of ITS have reduced effort and research support for both CBI and ITS.

I will show that these trends are premature. Advances in SLT and AuthorIT and TutorIT technologies based thereon **make it possible not only to duplicate human tutors in many areas but to do better.** Today, for example, few doubt we can build tutoring systems that teach facts as well or better than humans. "Flash cards", for example, could easily be replaced by computers – with more efficiency and certain results.

Today, I will go further:

a) I will show that AuthorIT makes it possible to create and that TutorIT now makes it possible to deliver highly adaptive (and configurable) tutoring systems that can do as good or better job on well-defined math skills.
b) I will show why and in what sense TutorIT tutorials can guarantee mastery of such skills.
c) I will show why TutorIT tutorials can be developed cost effectively – at half the cost of traditional CBI development.
d) I will show how TutorIT tutorials can gradually be extended to support the development and delivery of higher as well as lower order knowledge.
e) I will show why TutorIT tutorials can be expected to produce as good or better learning than most human tutors.

My paper is organized as follows:

1) I provide some background and summarize recent advances in knowledge representation and Structural Learning Theory (SLT) offering a theoretically rigorous, empirically sound foundation for building highly adaptive tutoring systems.
2) I first show why these advances make it possible to exceed human tutoring on well defined knowledge.
3) I then show how AuthorIT makes it possible to develop highly effective TutorIT tutorials at greatly reduced cost – even less than commercial development. I will also demonstrate how TutorIT math tutorials work and explain why they can do as good if not better job than most human tutors, how they can be configured at no additional cost to meet alternative needs – e.g., to serve as diagnostic systems, and how they can be used to reliably compare alternative pedagogies.
4) Finally, I talk about the future: What needs to be done to support ill-defined domains where higher order knowledge plays a central role? I show how AuthorIT and TutorIT can be extended to support adaptive tutoring on higher order learning – and why such tutoring systems may be expected to do as well, to even supersede human tutors.

II. BACKGROUND

In the 1960s, there was a disconnect in educational research and research in subject matter (math) education. Educational research focused on behavioral variables: exposition vs. discovery, example vs. didactic, demonstration vs. discussion, text vs. pictures, aptitude-treatment interactions, etc. (cf. Scandura, 1963, 1964a,b). Subject matter variables were either ignored or limited to such things as simple, moderate, difficult. Little attention was given to what makes content simple, moderate or difficult. Conversely, research in subject matter (e.g., math) education, focused primarily on content (reading, writing, arithmetic skills, algebraic equations, proof, etc.).

In same time period, instructional design focused on what was to be learned and prerequisites for same. Task analysis focused initially on behavior – on what learners need to do (Miller, 1959; Gagne, 1966). In my own work, this focus morphed into cognitive task analysis – on what learners must learn for success (e.g., Scandura, 1970, 1971, Durnin & Scandura, 1973). My parallel work in experimental psychology (Greeno & Scandura, 1966; Scandura & Roughead, 1967) in the mid 1960s added the critical dimension of behavior to the equation.

Structural Learning grew out of this disconnect, with the goal of integrating content structure with human cognition and behavior. Structural Learning Theory (SLT) was first introduced as a unified theory in 1970 (published in Scandura, 1971a). SLT's focus from day one (and the decade of research on problem solving and rule learning which preceded it) was on what must be learned for success in complex domains, ranging from early studies of problem solving and rule learning (Roughead & Scandura, 1968; Scandura, 1963, 1964a,b, 1973, 1977) to Piagetian conservation (Scandura & Scandura, 1980), constructions with straight edge and compass, mathematical proofs and critical reading (e.g., Scandura, 1977).

This research was focused on the following four basic questions [with their evolution from 1970 → Now]:

– Content: What does it mean to know something? And how can one represent knowledge in a way that has behavioral relevance?

  [1970: Directed graphs (flowcharts) → **Now: Abstract Syntax Trees (ASTs) & Structural Analysis (SA)]**

– Cognition: How do learners use and acquire knowledge? Why is it that some people can solve problems whereas others cannot?

  [1970: Goal switching → **Now: Universal Control Mechanism (UCM)]**

– Assessing Behavior: How can one determine what an individual does and does not know?

  [1970: Which paths are known → **Now: which nodes in AST are known (+), -, ?]**

– Instruction: How does knowledge change over time as a result of interacting with an external environment?

  [1970: Single level diagnosis & remediation → **Now: Multi-level inferences about what is known and what needs to be learned**]

Higher order and lower order knowledge played a central role in this research from its inceptions – with emphasis on the central role of higher order knowledge in problem solving (Scandura, 1971, 1973, 1977). Early SLT research also focused heavily on indentifying what individual learners do and do not know relative to what needed to be learned (e.g., Durnin & Scandura, 1974; Scandura, 1971, 1973, 1977).

Deterministic theorizing was a major distinguishing feature of this research (Scandura, 1971). I was focused, even obsessed with understanding, predicting and (in so far as education is concerned) controlling how individuals solve problems. Despite considerable training in statistics and having conducted a good deal of traditional experimental research (e.g., Greeno & Scandura, 1966; Scandura & Roughead, 1967; Scandura, 1967), I found unsatisfying comparisons based on averaging behavior over multiple subjects. I wanted something better – more akin to what had been accomplished in physics centuries earlier (cf. Scandura, 1974a).[2]

SLT was unique when introduced, and raised considerable interest both in the US and internationally (Scandura, 1971a, 1973, 1977). Literally hundreds of CBI programs based on SLT were developed later in the 1970s and early 1980s, and many sold for decades.

Nonetheless, ITS largely ignored this research and focused on later work in cognitive psychology (Anderson et al, 1990, 1993) and especially the Carnegie school of artificial intelligence based on production systems (esp. Newell & Simon, 1972).

By the mid-1970s, cognitive psychology also discovered the importance of content, often equating theory with alternative ways of representing knowledge. Research focused largely on what (productions or relationships) might be in learner minds and comparing fit with observable behavior. Experimental studies followed the traditional statistical paradigm.

Similarly, most CBI development was heavily influenced by Gagne's work in instructional design (1965), along with that of Merrill and his students, 1994). The restricted focus of Reigeluth's (1983, 1987) influential books on Instructional Design largely eliminated or obscured some of SLT's most Important features, most notably its focus on precise diagnosis and higher order learning and problem solving. With essential differences requiring significant study, the long and short of it is that other than our own early tutorials (which made small publisher Queue one of Inc Magazine's 100 fastest growing small businesses), SLT failed to significantly inform on-going research in either CBI or ITS. After the interdisciplinary doctoral program in structural learning I developed at Penn was eliminated in the early-mid 1970s, SLT became a little understood historical curiosity.

With recent publications in TICL, depth of understanding in ITS, CBI and SLT has increased in recent years (Mitrovic & Ohlsson, 2007; Paquette, 2007; Scandura, 2007), including their respective advantages and limitations (Scandura, Koedinger, Mitrovic & Ohlsson, Paquette, 2009). Advances in the way knowledge is represented in SLT has the potential of revolutionizing the way tutoring systems are developed, both now and in the future. SLT rules[3] were originally represented as directed graphs (e.g., Scandura, 1971a, 1973). Directed graphs (equivalent to Flowcharts) make it possible to assess individual knowledge. They have the disadvantage, however, of forcing one to make a priori judgments about level of analysis. They also make it difficult to identify subsets of problems associated with various paths in those graphs.

Having spent two decades in software engineering (e.g., Scandura, 1991, 1994 1995, 1999, 2001), it became increasingly apparent that a specific form of Abstract Syntax trees (ASTs) offered a long sought solution. ASTs are a precise formalism

[2] The deterministic philosophy I am proposing represents a major departure in thinking about how to evaluate instruction – in particular, it calls into question the usual measures used in controlled experiments. After understanding how TutorIT works, please see my concluding comments on this subject.

[3] I used the term "rule" rather extensively in behavioral research during the 1960s. Adopting the term "production" from the logician Post in the 1930s, Newell & Simon (1972) introduced the term "production rule" in their influential book on problem solving. Anderson later used of the term "rule" in ITS as synonymous with "production rule" (in production systems). Accordingly, it ultimately seemed best to introduce the term "SLT rule" to distinguish the two. Distinctive characteristics of SLT rules became even more important with my introduction of ASTs into SLT. In this context, ASTs represent a long sought solution to my early attempts at formalization in SLT (see Scandura, 1973, Chapter 3). The importance of ASTs in SLT, however, only gradually became clear to me after using the concept for some time in developing our software engineering tools -- despite the fact that ASTs had played a central role for years in compiler theory.

derived from compiler theory and that are widely used in software engineering. To date, ASTs have had almost no impact on knowledge representation, ITS or CBI. However, we will see that they do indeed have very significant advantages in SLT.

I have recently documented the current form of SLT in some detail (Scandura, 2007). Readers are encouraged to review the material therein on Knowledge Representation along with the published dialog on the subject which followed (Scandura, Koedinger, Mitrovic & Ohlsson and Paquette, 2009).

I focus in the next section on what is most unique about knowledge representation in SLT along with why and how it offers major advantages in developing adaptive tutoring systems.

III. Theoretical Advances: Well-Defined Knowledge

There have been three fundamental advances in SLT in recent years. First is in the way knowledge is represented. SLT rules were originally represented as directed graphs (Flowcharts). They are now represented in terms of Abstract Syntax Trees (ASTs). Second is formalization of a key step in Structural (domain) Analysis (SA), enabling the systematic identification of higher order SLT rules that must be learned for success in ill-defined domains. Third is the complete separation of SLT's control mechanism from higher order knowledge. These advances distinguish knowledge representation in SLT from all others, and have fundamental implications for building adaptive tutoring systems.

In this section we consider the first advance: SLT rules have long been used to represent to-be-acquired knowledge in well-defined domains. While retaining the advantages of directed graphs, representing SLT rules in terms of Abstract Syntax Trees (ASTs) offers a number of critically important benefits.

Not only do they offer a way to assess individual knowledge (as did directed graphs), but AST-based SLT rules also provide a perfectly general way to automatically both generate test problems and the solutions to those test problems. As we shall see, they also make it possible to simultaneously represent knowledge at any number of levels of analysis.

Precision.– The major reason adaptive tutoring systems have been so difficult and expensive to develop is that pedagogical decision making has been so time consuming and expensive. This is equally true whether tutoring systems are based on traditional CBI (cf. Paquette, 2007) or ITS (cf. Mitrovic & Ohlsson, 2007).

In CBI the focus is on what must be learned. Better CBI systems invariably are based on some combination of hierarchical and/or relational analysis. Hierarchical representations have an important advantage: Hierarchies inherently arrange content in the order in which content must be learned. Content higher in a hierarchy necessarily incorporates lower order content, a fact that has direct and important implications for both testing and teaching.

The problem is twofold:

(1) not everything can be represented hierarchically using current decomposition methods (Scandura, 2007) and

(2) informal hierarchical representation is not sufficiently precise to automate decision making without direct attention to the meaning of the content.

I don't want to don't have time to repeat here what has already been published. On the other hand, I must call special attention to one key idea, an idea that makes it possible to develop adaptive tutoring systems that can both: a) be developed at lower cost and b) guarantee learning.

Specifically, Structural (domain) Analysis (e.g., Scandura, 2007) makes it possible not only to represent all behavior hierarchically, but to do so precisely that inherent relationships are exposed.

It is well know known that many ideas can be refined into components or categories. Components and categories are fundamental: Component refinements involve breaking sets into to their elements. Category refinements involve breaking sets into subsets. For example, the set of animals can be refined into elements – individual animals in the set. The set of animals also can be refined into categories: dogs, cats, whales, etc.

Consider column subtraction: We begin with a subtraction problem. Subtraction problems typically are refined first into elements, the columns that make up a subtraction problem. (Because the number of columns in a subtraction problem may vary, I have called this variation a "prototype" refinement, wherein each prototype, or column, has the same structure.) Columns, in turn, may be refined into categories, columns where the top number is greater than or equal to the bottom number and columns where the top number is less than the bottom number.

The same idea applies generally: Consider a "house". Houses consist of sets of rooms, room elements. Rooms in turn can be categorized by their size, or their use, or by any number of other distinctions.

As detailed below, component and category refinements have direct counterparts in corresponding solution procedures. Again, consider column subtraction. Here, the initial procedural refinement is a Repeat-Until loop. Loops in procedures correspond precisely to Prototype refinements in data: Compute the answer to each column in turn until there are no more columns. The next procedural refinement is an IF..THEN selection. Selection refinements in procedures correspond to Category refinements in data. In subtraction, different processes are required when the top number is greater than or equal to the bottom number and when this is not the case.

Unfortunately, component and category refinements are not sufficient. Other kinds of "refinements" involve (more general) relationships – for example, whether the top digit is greater than or equal to the bottom digit. Mating involves a relationship between two animals – male and female.

Simple relationships are fine when they are immediately understandable and unambiguous. In many cases, however, they are not. None of us, for example, would any problem writing the numeral "5". Ask most four or five year olds, however, and the story is likely to be very different. Writing the numeral "5" requires a precise set of constructions involving straight and curved line segments.

Relational models can easily represent the relationships between such line segments. Indeed, <u>everything</u> can be represented in terms of relationships. The problem is twofold. The number of relationships increases rapidly as domains become increasingly complex. In complex domains, relationships on relationships can extend geometrically without bound.

Relational representations suffer from an additional problem (beyond the sheer number of relationships). Whereas component and category refinements may be repeated indefinitely, this is <u>not</u> possible with (non-unary) relationships. Every relationship (relational refinement) must be considered anew. There is no systematic way to represent given (non-unary) relationships in terms of simpler elements.

Knowledge representation using ASTs solves this problem. <u>There is a fundamental mathematical equivalence between relations and functions</u>. Each and every relationship can be represented by at least one function, or procedure, having its own inputs and outputs. For example, relationships between straight and curved line segments comprising the numeral "5" can be viewed as a procedure operating on such segments. These procedures in turn can be refined as the originals.

Why is this important? Consider the following. If we subject Column Subtraction to Structural Analysis, we are going to end up with terminal elements requiring such things as the child's ability to write the numeral "5" (and "0", "1", "2", …). No matter what is being learned there will always be things that learners must know on entry. Young children, for example, learn early on to do such things as write the numeral "5". What is being learned here is not a relationship. Rather, it is an SLT rule that takes line segments as input and generates the numeral "5".

Prerequisite SLT rules, in turn, can be refined as any other. The refinement process can be repeated indefinitely. <u>No matter how complex the subject matter, or how naïve the target population, it is always possible to represent the knowledge necessary for success in hierarchical form</u>. The introduction of what I have called "dynamic" refinements, along with component and category refinements, closes the loop. It is now possible to represent what needs to be learned in any domain in whatever detail may be necessary (and desirable).

NOTE 1: It is worth noting incidentally that representing relationships as functions is equivalent in software engineering to introducing the notion of a "callback". Just as one may introduce functions operating on parameters in a dialog box, one can introduce functions generating outputs from inputs in a relationship.

NOTE 2: It might appear that arbitrary refinement may be as, if not more demanding than knowledge engineering in ITS. Identifying possible (correct and/or error) productions, however, not to mention learning mechanisms, can be very challenging and open ended. On the other hand, the process of Structural Analysis (SA, is highly systematic with a definitive end point. In addition to learning how to perform SA using AuthorIT's AutoBuilder component (see below), the main requirements for an author are the ability to perform the skill in question and reasonable insight into what must be learned for success. Working under my direction a single programmer familiar with AuthorIT and TutorIT has been making TutorIT tutorials ready for field testing at a rate of at least one per month. Only a small portion of this time has involved representing to-be-acquired knowledge. Most has been devoted to laying out interfaces and associated media.

<u>For those not mathematically inclined, all this may seem like a technical truism with little practical significance</u>. In fact, however, this technical truism has fundamental practical significance. <u>AST hierarchies provide a perfectly general way to define pedagogical decisions</u>. All pedagogical decisions in SLT can be based entirely on the structure of to-be-learned SLT rules. This can all be done independently of content semantics.

Indefinite refinement makes it possible to define what needs to be learned with whatever precision may be necessary to make contact with knowledge available to any population of learners, no matter how naïve they might be initially.

Full hierarchical representation makes it possible to quickly determine the status of any individual's knowledge at each point in time (relative to a SLT rule hierarchy), and to provide the instruction necessary to advance. Given full analysis, empirical research (e.g., Scandura, 1970, 1971, 1973, 1974a, 1977' Durnin & Scandura, 1973) demonstrates that testing on a single test item is sufficient to determine whether a learner has mastered any given sub tree in an SLT rule.

It is not always feasible, however, nor necessary to undertake complete analysis. Nonetheless, even incomplete hierarchical analysis is better than none. Incomplete hierarchies provide a beginning – a starting point that can be improved incrementally as time, resources and the importance of any particular tutoring system demands.

NOTE: Curriculum standards specifying prerequisites, concepts to be learned and the order in which they should be acquired may serve as a starting point. Generally speaking, however, our experience is that they do not normally go nearly far enough in identifying what must be learned for success.

It is always <u>possible to build effective tutoring systems by introducing a safety factor</u> (Scandura, 2005) – as engineers do in designing a bridge. Instead of requiring a single success corresponding to any terminal node (in an SLT rule hierarchy) one can require any number of successes. This makes it possible in principle to <u>guarantee learning</u>.

<u>Efficiency of Development</u>.– A major limitation of adaptive tutoring systems is that they have been hard to build.

Identifying knowledge is only one part of the process. Defining (and implementing) pedagogical decisions in traditional ITS – what to test or teach and when – is the most expensive, time consuming and error prone tasks required (Mitrovic & Ohlsson, 2007; Koedinger & Ohlsson, 2009). This perhaps is the primary reason so few truly adaptive tutoring systems exist despite many years of university and federal support.

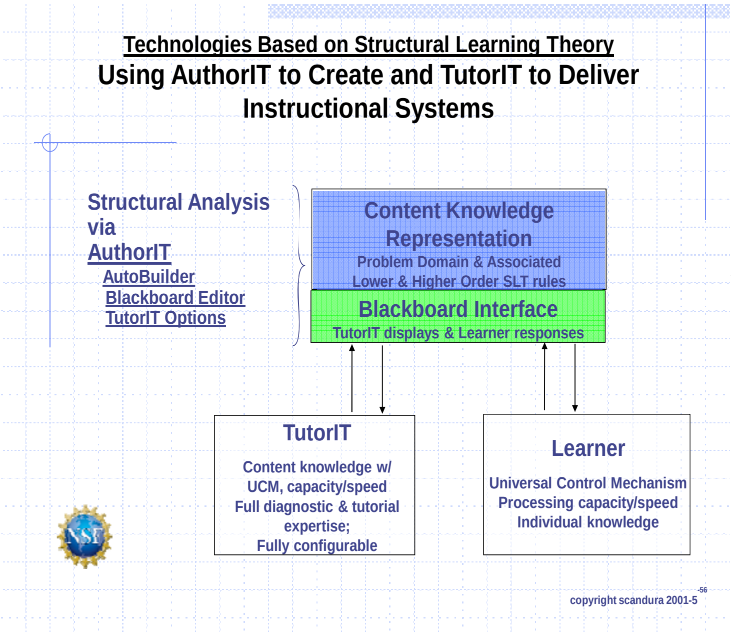
By way of contrast, I will show how TutorIT makes all pedagogical decisions automatically –based entirely on the hierarchical structure of SLT rules representing what is to be learned.

Hierarchical representation has a further not inconsequential benefit. It is easy to define any number of pedagogical theories as to how best to promote learning. Specifically, I will show how TutorIT can easily be configured to deliver instruction in accordance with a variety of pedagogical philosophies. In all cases, TutorIT effectively eliminates the need to program pedagogical decisions (Scandura 2005, 2007, 2009). Cost savings have been estimated at between 40 and 60% (cf. Foshay & Preese, 2005, 2006; Scandura, 2006a,b).

In short, guaranteed results at lower cost – a combination that should be hard to resist.

IV. Current Status of TutorIT: Guaranteed Learning and Lower Cost

Our AuthorIT authoring and TutorIT delivery systems currently support the development and delivery of well defined knowledge (Scandura, 2005). The long term goal, however, is to realize the full potential of SLT (Fig. 1).



Technologies Based on Structural Learning Theory

Using AuthorIT to Create and TutorIT to Deliver Instructional Systems

Structural Analysis via AuthorIT

AutoBuilder

Blackboard Editor

TutorIT Options

Content Knowledge Representation

Problem Domain & Associated Lower & Higher Order SLT rules

Blackboard Interface

TutorIT displays & Learner responses

TutorIT

Content knowledge w/ UCM, capacity/speed

Full diagnostic & tutorial expertise;

Fully configurable

Learner

Universal Control Mechanism

Processing capacity/speed

Individual knowledge

copyright scandura 2001-5

Fig. 1. Technologies based on Structural learning theory.

A. Given a well defined problem domain, AuthorIT includes the following:

1) AutoBuilder, a tool for systematically representing knowledge as an SLT rule, including both the procedural Abstract Syntax Tree (AST) and the AST data structure on which it operates. Procedural ASTs in AutoBuilder are visually represented as Flexforms (below). Each node in a Flexform represents a specific part of the to-be-acquired knowledge.
2) Blackboard Editor, a tool for creating and laying out schemas representing problems in the domain. Blackboard serves as the interface through which learners and TutorIT interact.
3) AutoBuilder also is used to assign instruction, questions, positive feedback and corrective feedback to individual nodes in the Flexform. This information may include text, graphics, sound and/or other supporting media.
4) Options, a dialog (tool) used to define how TutorIT is to interact with learners. Options include variations on delivery modes ranging from highly adaptive to diagnostic, to simulation to practice.

B. TutorIT takes the above produced with AuthorIT and interacts with learners as prescribed in the Options Tool. I will show how TutorIT's adaptive mode works below. But, first let's review the development process.

Representing Well defined Knowledge.– TutorIT development begins by representing to be learned knowledge as an SLT rule. As required by SLT (Scandura, 2005), AutoBuilder makes it possible to represent knowledge with arbitrary degrees of precision.

Each node in the Flexform represents to-be-acquired knowledge at a specific level of abstraction. For example,

A. "*Borrow_and_subtract_the_current_column*" to the right of the first "*ELSE*" in Fig. 2 (Appendix A) represents the knowledge necessary for computing the difference in any column when the top digit is less than the bottom digit.

B. Subordinate nodes like "*Borrow_from_next_column*" provide increasingly more specific information.

Parameters of these operations representing data on which these operations act also are arranged hierarchically. Operation A, for example, operates on "Prob" and "CurrentColumn". "Prob" represents an entire subtraction problem. "CurrentColumn" represents columns in such problems. Operation B also includes "ReducedTop", "Slashtop", "CurrentBorrowColumn" and "BorrowedDigit".

In this context, AuthorIT's AutoBuilder component imposes consistency requirements on successive refinements. These requirements are designed to ensure that the behavior of children in each refinement is equivalent to the behavior of the parent. Operation A, for example, operates on each current column without a computed difference and generates the current column with the correct difference. The nodes immediately below Operation A provide more detail as to the intermediate steps and decisions. Otherwise, however, they produce the same result. The behavior is equivalent.

In general, higher level nodes may operate on more highly structured parameters. For example, CurrentColumn represents entire columns, including the column itself and the top, bottom, difference and borrow digits in that column. Corresponding lower level child nodes operate on simpler parameters, like BorrowedDigit. The behavior of higher and lower level operations (i.e., nodes), however, is expected to be equivalent (to produce equivalent results).

Defining Problem Schemas.– Once the Flexform has been fully implemented (and tested using AuthorIT's build in Interpreter/Visual Debugger), the next major step is to define problem schemas that collectively exercise all nodes in the Flexform. For example, a subtraction problem with all top digits greater than or equal to the bottom ones will not exercise Flexform nodes involving regrouping (or borrowing). Problem schemas are defined and laid out in AuthorIT's Blackboard Editor as shown in Fig. 3 (Appendix A).

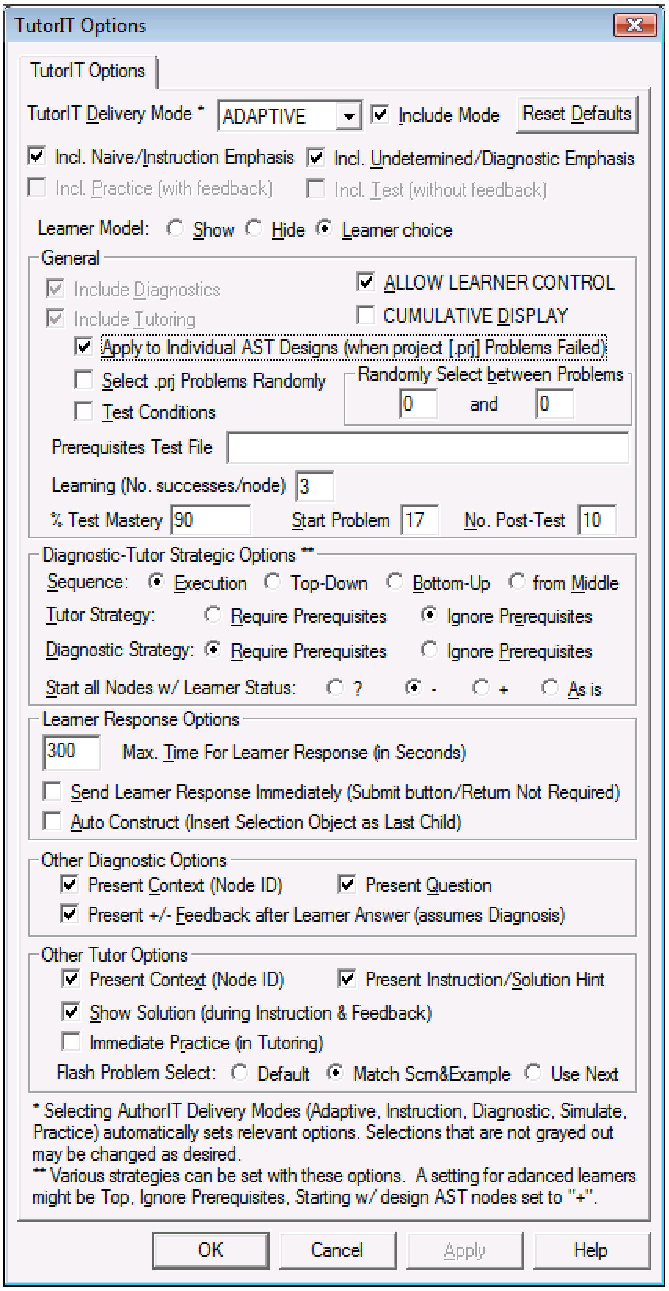
TutorIT Options Tool.– The Tutor Options Tool, currently a dialog in AuthorIT, is used by authors to define/configure alternative learning modes. The first decision an author must make is to decide which of the basic TutorIT Delivery Modes to include: ADAPTIVE, INSTRUCTION, DIAGNOSTIC, SIMULATION or PRACTICE. The Options Tool in Figure 5 is set to ADAPTIVE mode. Authors also can make DIAGNOSTIC, INSTRUCTION, SIMULATION, and PRACTICE modes available in TutorIT by selecting desired modes for TutorIT Delivery Mode in the dialog. ALLOW LEARNER CONTROL also is an option. In short, TutorIT makes it possible to compare different pedagogies on even terms.

TutorIT.– The Flexform associated with a skill represents what is to be learned in an arbitrarily precise manner. The Flexform design (in blue) also includes HLD code (in green) which is interpretable by TutorIT. The design Flexform acts like a structured database, including all information needed by TutorIT to provide a wide variety of delivery modes. In addition to to-be-learned operations and decisions, Flexforms include: a) a modular executable implementation of each terminal (Fig. 2), b) questions, instruction, feedback and corrective feedback associated with specific nodes in the Flexform, c) problem schemas (which serve as input to the Flexform) problem schemas laying out the kinds of problems to solved (Fig. 3), and d) TutorIT options specifying how TutorIT is to make its decisions (Fig. 4).

The way TutorIT operates depends on how it is configured in the Options Tool. I concentrate here primarily on Adaptive mode. Other options, such as Diagnostic and Instruction, are special cases or restrictions. TutorIT in Adaptive mode automatically selects nodes that quickly pinpoint what a learner does and does not know at each stage of learning.

Learner Model.– TutorIT takes the above Flexform files as input and first creates a Learner Model representing what the learner initially knows or is assumed to know about the to-be-learned Skill. The Learner Model is normally displayed in a tree view with each leaf marked with a "+", "-" or "?" corresponding to a (blue or actionable) node in the Flexform (e.g., see Fig. 5, in the Appendix A).

The left side of Figure 5 shows the Blackboard interface through which TutorIT interacts with the learner. The Learner Model for a student just beginning Column Subtraction is shown on the right side of Figure 5.



TutorIT Delivery Mode * ADAPTIVE ☑ Include Mode Reset Defaults

☑ Incl. Naive/Instruction Emphasis ☑ Incl. Undetermined/Diagnostic Emphasis

☐ Incl. Practice (with feedback) ☐ Incl. Test (without feedback)

Learner Model: ○ Show ○ Hide ● Learner choice

General

☑ Include Diagnostics ☑ ALLOW LEARNER CONTROL

☑ Include Tutoring ☐ CUMULATIVE DISPLAY

☑ Apply to Individual AST Designs (when project [.prj] Problems Failed)

☐ Select .prj Problems Randomly

☐ Test Conditions

Randomly Select between Problems 0 and 0

Prerequisites Test File

Learning (No. successes/node) 3

% Test Mastery 90 Start Problem 17 No. Post-Test 10

Diagnostic-Tutor Strategic Options **

Sequence: ● Execution ○ Top-Down ○ Bottom-Up ○ from Middle

Tutor Strategy: ○ Require Prerequisites ● Ignore Prerequisites

Diagnostic Strategy: ● Require Prerequisites ○ Ignore Prerequisites

Start all Nodes w/ Learner Status: ○ ? ● - ○ + ○ As is

Learner Response Options

300 Max. Time For Learner Response (in Seconds)

☐ Send Learner Response Immediately (Submit button/Return Not Required)

☐ Auto Construct (Insert Selection Object as Last Child)

Other Diagnostic Options

☑ Present Context (Node ID) ☑ Present Question

☑ Present +/- Feedback after Learner Answer (assumes Diagnosis)

Other Tutor Options

☑ Present Context (Node ID) ☑ Present Instruction/Solution Hint

☑ Show Solution (during Instruction & Feedback)

☐ Immediate Practice (in Tutoring)

Flash Problem Select: ○ Default ● Match Scrn&Example ○ Use Next

* Selecting AuthorIT Delivery Modes (Adaptive, Instruction, Diagnostic, Simulate, Practice) automatically sets relevant options. Selections that are not grayed out may be changed as desired.
** Various strategies can be set with these options. A setting for adanced learners might be Top, Ignore Prerequisites, Starting w/ design AST nodes set to "+".

Fig. 4. TutorIT Options Tool used by authors to define/configure alternative learning modes. Currently set to ADAPTIVE mode. Other choices allow for further customization.

Given the above information, TutorIT operates as follows. All decisions are based entirely on the structure of the content to be learned, independently of content semantics.

1. TutorIT selects a problem.
2. TutorIT then selects a (blue) node in the Flexform (or Learner Model). Only nodes that are exercised by the

selected problem are eligible for selection. Selections otherwise are made according to priorities set in TutorIT's Options Tool (Fig. 4).

3. TutorIT executes the Flexform using its built in interpreter. The subtree defined by the selected node (in the Flexform) automatically generates a sub-problem of the problem schema and also its solution.

NOTE: As below, we will want TutorIT to also support the case where TutorIT must generate (new) solution Flexforms from higher and lower order SLT rules. SLT's UCM will play a central role in this context. Currently, TutorIT only supports chaining two or more SLT rules as in current expert systems.

4. TutorIT displays the sub-problem on TutorIT's blackboard (see the left side of Fig. 5).
5. If a node is marked "-", TutorIT provides instruction. If marked "?", TutorIT presents a question to determine its status. TutorIT skips nodes marked "+" unless the node is an automation node. Automation nodes require a faster response (higher level of expertise).

NOTE: These questions and instructions, as well as positive and corrective feedback, may consist of simple text, voice and/or media consisting of Flash, audio-visual or other files.

6. TutorIT compares the learner's response with the correct answer, which is automatically generated by TutorIT.
   a. If the status was "?" and the learner gets the correct answer, positive feedback is given and the node is marked correct (assigned a "+"). If incorrect, TutorIT provides corrective feedback and the node is marked with a minus ("-").
   b. If the status was "-", instruction is given and the node is marked "?". After instruction, it is impossible for TutorIT to know for sure that the learner has actually learned what was taught.

NOTE: The learner must meet timing requirements if the node is an automation node requiring a higher level of skill.

7. In addition to determining the learner's status on individual nodes, TutorIT also infers what the leaner knows with respect to nodes dependent on the current one:
   a. If the learner gives an incorrect response, TutorIT reliably assumes that any (higher level) node dependent on it also should be marked unknown. TutorIT marks such nodes accordingly.
   b. Conversely, if the learner gives the correct response, TutorIT reasonably assumes that the learner also knows those lower level nodes on which it depends. In short, TutorIT not only compares learner responses on sub-problems corresponding to individual nodes but also quickly infers what the learner knows about nodes dependent on it.

TutorIT can be configured with various "safety factors" to ensure learning. For example, one can set the Options Tool to require learners to demonstrate mastery on every node, not just once but any specified number of times (see "Learning (No. successes/node"). After learning, TutorIT can be set to require any specified level of success on practice problems.

To date, we have developed TutorIT tutorials for Column Addition, Column Subtraction, Column Multiplication and Long and Short Division along with five levels for each of the Basic Facts: Addition, Subtraction, Multiplication and Division. Operations on Fractions also are in progress. Please see *www.TutorITmath.com* for latest availabilities.

In each case, TutorIT takes problem schemas as laid out in the Blackboard Editor as input. It automatically generates problems, actually sub-problems, as needed for diagnosis and remediation. Nodes are selected so as to enable TutorIT to quickly pinpoint what each individual does and does not know at each point in time, and to provide precisely the information (instruction) needed when needed to progress in optimal fashion.

All this is done dynamically during the course of instruction as might a human tutor. The main difference is that TutorIT does this in a highly disciplined manner. All decision making is done automatically based entirely on the structure of the to-be-acquired knowledge. Semantic independence dramatically reduces the effort required to create adaptive tutoring systems.

The hierarchical representation of knowledge (in Flexforms) has important implications for both efficiency and effectiveness. As above, TutorIT makes direct inferences not only with respect to individual nodes but to dependent nodes as well. For example, if a student gets a problem associated with one node correct, then TutorIT can reliably assume that the student also knows all of the lower level nodes on which it depends. For example, if a child can subtract columns involving borrowing or regrouping, one can reasonably assume that the child can also subtract successfully when there is no regrouping. On the other hand, if a child cannot subtract a column that does not involve regrouping, one can be quite certain, he or she cannot subtract when regrouping is required. In short, success on a node implies success on all subordinate nodes. Failure implies failure on all superordinate nodes. The result is very efficient diagnosis and instruction.

Unlike most teachers, TutorIT can be unusually effective because it benefits from careful pre-analysis. We have put a considerably amount of effort into our TutorIT Math skill tutorials – far more than what goes into writing a text book for example. The level of analysis in our current prototypes can and will be further improved as a result of field testing. Even in their current state, however, TutorIT Math skill tutors benefit from considerably more analysis than most teachers are capable. And, this analysis can further be improved incrementally.

On the other hand, of course, a good human tutor generally will be more attuned to motivational factors. We expect TutorIT tutorials to get better and better over time as a result of feedback. Nonetheless, they are designed for specific purposes and may never achieve the flexibility of a good human tutor who has spent years both learning math and how to motivate children to learn in a wide variety of real world situations. In short, there likely will always be some things that a good human can do better than TutorIT – as well as the converse. Having said this, the choice is not one of either or. Rather it is a question of how best to use both to maximize learning.

Importance of Prerequisites.– One might argue that just because a student solve one subproblem (associated with a given node) does not necessarily imply that he or she can do this with all such subproblems. Indeed, this is correct. As pointed out in my earlier description of TutorIT (Scandura, 2005) a single test will only be sufficient when analysis is complete – when all terminal nodes are as we say atomic. Success on one instance in this case implies success on all instances of the same type with unusually high degrees of reliability (cf. Scandura, 1971a, 1973, 1977). Having said, this just as a good bridge designer builds in a safety factor, an author can easily do the same with TutorIT. TutorIT can be required to demand a higher level of performance by simply changing a setting in the Options Tool to require any number of successes on each node (before mastery is assumed).

Criteria may be set so as to actually guarantee learning. Any learner who enters with pre-specified prerequisites, and who completes a given TutorIT tutorial will be definition have mastered the skill in question. There is no other way a student can complete a TutorIT tutorial. He or she must meet pre-specified criteria set by the author or TutorIT tutoring will continue until they are met.

Notice that prerequisites play an essential role in the process. Prerequisites correspond precisely to atomic or terminal nodes in Flexform knowledge representations. Some prerequisites are so simple that they can safely be assumed on entry. For example, the ability to read and write numerals. Entry with respect to other prerequisites, however, may be less certain. Any child presumed to be ready for long division would almost certainly have to know the multiplication tables and how to subtract. Similarly, no would want to teach column subtraction unless a child already knew how to count.

The basic question in this context is how one make contact with learner's who have not mastered such prerequisites? For example, how to teach column subtraction to a child who cannot write or recognize numerals (e.g., "5", "3"). SLT support for indefinite refinement offers a unique solution to this problem. One is not forced to introduce non-decomposable relationships. Instead, each such prerequisite can be represented as an equivalent SLT rule with its own domain and range. As above, for example, the numeral "5" can be viewed as an SLT rule for constructing the numeral from more basic line segments. Most important, SLT rules representing prerequisites can be refined further just as any other.

One further point. Let's turn this argument on its head. The difficulty of any task, or to be learned skill depends not on just the skill itself. Rather, it depends on the nature of the prerequisites that may be assumed available. We hear a lot, for example, about the benefits of using sophisticated calculators in education (e.g., TI's Nspire family). Clearly, if one has a calculator, computational issues take a back seat. It is far easier to learn to evaluate arithmetic computations with a calculator than without.

NOTE: Along with most mathematics educators I would argue nonetheless that computational abilities are essential irrespective of the presence or absence of a calculator.

On the one hand mastering Nspire can be can subjected to the same kind of analysis we are talking about here. And, TutorIT could equally well be used to provide the necessary instruction. On the other side of the coin, one can start with the assumption that learners can already use of such tools as Nspire – as prerequisites on entry. In this context, to-be-analyzed problem domains will be very different.

Instead of computational skills, the focus is more likely to be on problem analysis. Given a description of a situation, for example, how can it be formulated in terms of mathematical expressions? Having created such an expression, one can plug in the numbers and click to get the solution. In a similar manner, the more comprehensive the skills one can assume the more sophisticated the knowledge one can teach. The general truism to be taken from this analysis is not whether basic skills are important but rather that the more basic skills one has mastered, the more one has to build one. This is true whether in mathematics or in any other subject.

NOTE: Representing reality in terms of mathematical expressions is one of six basic process abilities in mathematics. These were first introduced in Chapter 1 of my book on Mathematics: Concrete Behavioral Foundations (Scandura, 1971b, pp. 3-64). The six abilities were organized as three bidirectional pairs: Detecting regularities and its opposite of constructing examples of regularities, understanding mathematical representations (e.g., expressions) and its opposite of creating mathematical expressions and deduction and its opposite axiomatization.

Configuring TutorIT.– The ease with which TutorIT can be customized adds another important dimension. In addition to ADAPTIVE mode, the Options Tool also supports DIAGNOSTIC, INSTRUCTION, SIMULATION and PRACTICE modes. Authors may also allow learner Control, in which case the learner may decide on which items to be questioned or to receive instruction.

Each basic delivery mode comes with some mandatory settings. Other options enable authors to better control the way content is delivered.

At the most basic level, for example, a student might already have been exposed in varying degrees to the knowledge being taught. In this case, TutorIT cannot know what the learner knows on entry. In so far as TutorIT is concerned, the learner enters essentially as a blank slate. Conversely, if a student has had no exposure to the content, TutorIT might start with nodes marked "-""-", or unknown. In this case, TutorIT will initially be biased toward instruction.

In this case, "*Start all Nodes with the Learner Status*" in AuthorIT's Options Tool (see Fig. 4) might be set to "?" or "-" depending on prior student exposure to the content,. The author also has the choice of allowing teachers or students to make the choice for individual students or to require it for all.

In the undetermined state, TutorIT starts tutoring each child by marking each node in the Learner Model (see below) with a "?". This signifies that TutorIT does not (yet) know whether or not a learner has mastered the knowledge associated with that node. After the learner responds, TutorIT provides corrective or positive feedback as appropriate – and updates the Learner Model as above – "+" for success or "-" for failure.

Other options provide finer levels of control. For example, an author might require that instruction be given only when ALL prerequisite nodes have been mastered (marked "+"). Alternatively, the author might want to place more emphasis on self-discovery. Here, the author might choose the *Ignore Prerequisites* option for *Tutor Strategy*. In this case TutorIT will provide hints/scaffolding (i.e., instruction) even when the learner's status on lower level nodes is unknown.

More generally, the author has a wide variety of options making it possible to accommodate a wide variety of pedagogical biases – or should I say "instructional theories". Available options support a wide variety of instructional philosophies – ranging from highly directive instruction to open ended discovery including completely self directed learning.

Comparison and Benefits.– Like other ITS or CBI, TutorIT Math tutors are highly reliable. They never tire. They never make mistakes – excepting bugs one may have missed.

Unlike other CBI (or ITS), however, TutorIT Math tutors are designed so that any learner who enters with pre-specified prerequisites and who completes a given tutorial will necessarily have mastered the skill in question.

Whether these results are realized with actual students depends on the following assumptions: a) that we have in fact identified an SLT rule for correctly performing a ranges of basic math skills with sufficient precision to have identified essential prerequisite skills (terminals in Flexform used to represent SLT rules), b) that learners demonstrate mastery of those prerequisites on entry and c) that students complete the TutorIT tutorial – the only way a student can do this is to have demonstrated mastery on the skill being taught to whatever criterion the author has prescribed (in the Options Tool).

In effect, what the student learns and whether or not a student who completes a tutorial actually learns the skill is not a question to be determined empirically. Rather, the proof will be in such things as how long it takes, whether students are sufficiently motivated to complete the tutorial, and generally what might be done to make the tutorial even better (e.g., more efficient and/or motivating for students, etc.). Given the way TutorIT tutorials are developed, improvement will occur incrementally as feedback suggests and as resources allow.

Toward this end, we are just now beginning field testing by offering free trials. Anyone, a school, tutoring center, or home can get free individualized tutoring on whatever skills are currently available (now 5 levels for reach of the basic facts and the basic whole number algorithms for addition, subtraction, multiplication and division).

At the same time, the AuthorIT/TutorIT system dramatically reduces development costs. As above, we have already developed a range of TutorIT tutorials at a fraction of the costs of ITS development. These tutorials focus on very specific identifiable skills. Guaranteed learning is restricted specifically to those skills. Nonetheless, TutorIT tutorials developed to date also include instruction pertaining to meaning. I refer here to the kinds of instruction commonly included in textbooks and classroom instruction.[4]

There is no guarantee having gone through a given TutorIT tutorial that students will necessarily also master this supplemental material – material that is normally included (but also not guaranteed) in classroom instruction. The question of whether and to what extent this supplemental instruction benefits students is an empirical one. Given TutorIT's focus on doing what it can do better and more efficiently than a human (or any other means of transmittal), this question also is of secondary importance. Current TutorIT tutorials are designed to support classroom instruction not to replace what a good teacher can (or should) do.[5]

How can that be – better results at lower cost? The answer lies in the very close relationship between knowledge representation, on the one hand, and diagnostic and remedial actions on the other. On the one hand, arbitrary refinement allows for indefinite precision. Tutoring can be guaranteed. Learners who enter with predetermined prerequisites and who complete a given TutorIT tutorial will by definition demonstrate mastery of defined skills.

The same structural relationships that make it possible to provide efficient, highly targeted adaptive instruction also eliminate the need to program pedagogical decisions. While estimates may be based on slightly different assumptions, the bottom line is that roughly half of all development costs can be eliminated (cf. Foshay & Preese, 2005, 2006; Scandura, 2006a,b). It is not necessary to independently program

[4] It is not that one could not target meaning as such. It is simply that doing so would require further analysis. For example, TutorIT Column Subtraction is based on a detailed analysis of what must be learned to perform column subtraction – with learning guaranteed when a student completes the tutorial. This tutorial also includes instruction describing and graphically illustrating a concrete model of what is being done step by step (e.g., when one borrows during subtraction). The difference is that we have not undertaken a systematic analysis of what would need to be learned to ensure that a student is able to demonstrate the meaning associated with any given problem, or the reverse to construct a physical model corresponding to any given subtraction problem. We could! We just haven't, nor has any text book we know of as well. "Dienes blocks" developed in the 1960s by an old colleague of mine were designed precisely for this purpose.

[5] While TutorIT Math is not sufficiently complete to cover all that is in a typical textbook. Other than background reading and the like, it is an open question as to whether there are specific skills in a math textbook that could not be done as well (or better) in TutorIT.

diagnostic and instructional logic as in developing other adaptive tutoring systems.

In comparison with other approaches, AuthorIT and TutorIT offer three major benefits:

a) Better results on well defined tasks than even human tutors due both to more complete analysis (than most humans are capable of) and to highly effective and efficient tutoring. The latter derives from TutorIT's optimized pedagogical decision making. More complete analysis and optimized decision make it possible under carefully prescribed conditions to actually guarantee learning. The way things are set up there is essentially no way a student can complete a TutorIT math tutorial without mastering the skill. The question is not learning as such but whether a student is motivated to complete a given tutorial, a very different question requiring a very different answer.

b) Greatly reduced development costs because all TutorIT decision making is predefined. All diagnosis and testing is automatic and based entirely on the structure of the to-be-learned knowledge. While we have not kept actual figures on development costs, they are by definition an order of magnitude less than that required in ITS development. TutorIT tutorials ready for field testing have been completed by myself with the assistance of approximately one full time person for a year. With an experienced team and further maturity of AuthorIT and TutorIT, development costs may be expected to go down gradually.

c) Furthermore, pedagogical decision making is fully configurable. TutorIT can easily be configured to provide adaptive tutoring customized for different learners both individually and by population. TutorIT also can be configured to provide highly adaptive diagnosis, to provide practice or to serve as a performance aid. Configuration consists entirely of making selections in an Options dialog – all without any programming or change in the knowledge representation.

NOTE: The notion of (content) domain independent instructional systems is not entirely new. It is not difficult, for example, to construct CBI systems that support specific categories of learning, such as those defined by Gagne (1985). The closest analog is probably Xaida (e.g., see Dijkstra, Schott, Seel & Tennyson,1997). TutorIT takes a major step forward in this regard by providing tutoring support for ANY well defined content. This not only includes all Gagne's categories of learning, for example, but any combination thereof.

The bottom line is that TutorIT is another significant step forward in automation. TutorIT provides another case where computers can do things better than a human – this time in adaptive tutoring. As more and more tutorials are developed TutorIT can gradually taking over tasks previously done by humans – not just in math skills but ultimately with any well defined skill. TutorIT tutorials will gradually take over for one reason: Not because they are approaching what humans can do but because they can do some jobs better than humans.

TutorIT, of course, will not eliminate the need for good teachers any more that good computational tools have eliminated the need for people who use them. TutorIT tutorials will enable teachers to concentrate on things they can do better. TutorIT automation will be an on-going and continuing process. Our children and our country will be the main beneficiaries. If you might be interested in contributing to this effort please let me know at *scandura@scandura.com*.

V. Critical Advances in Current SLT Theory

Analyzing Complex Domains.– It might seem we are done! Given any domain, we can always use AuthorIT's AutoBuilder component to systematically identify what needs to be learned for success – with whatever degree of precision may be necessary or desired. The rest follows automatically. TutorIT takes the representation produced (including display layouts and associated media) as input and automatically delivers instruction as prescribed.

While theoretically possible, identifying what must be learned as an SLT rule is not necessarily easy. It can be very difficult, practically impossible to identify a single, integrated SLT rule that represents the knowledge needed to master complex domains. This is not simply having to compromise as regards completeness.

It would be impractical if not impossible to directly identify what must be learned to prove all known theorems in mathematics, or to specify how to write a beautiful poem (given some topic or idea). As those engaged in ITS development know, identifying what needs to be learned in high school algebra already poses a difficult task (Ritter, 2005).

By way of contrast, high level relational models are relatively easy to create (cf., Scandura, 1973; Hoz, 2008). Relational models, however, lack precision – and complexity increases rapidly. Both constraints place significant limits on effective tutoring. Equally important, pedagogical decisions based on relational models can be very difficult. Pedagogical decision making depends inextricably on content semantics, thereby increasing both development and evaluation costs.

In SLT, it might appear that one can avoid this problem by simply introducing a finite set of SLT rules. For example, instead of one SLT rule as above, why not simply add new SLT rules? Certainly, one can do this. Doing so, however, does not solve the fundamental problem. Given any non-trivial domain, it is impossible to directly identity everything a learner should know. This fact has been a central tenet in SLT from its inceptions (cf. Scandura, 1971a). It was the primary motivation for introducing higher order rules.

ITS systems approach this problem from a very different perspective. Beginning with Newell & Simon's (1972) influential work on problem solving, the focus has been on identifying sets of productions corresponding to what might be in human brains. ITS knowledge engineers work with subject matter experts to identify condition-action pairs, or productions representing relevant knowledge. Productions collectively are expected to be sufficient for solving arbitrary problems in a given domain.

Identifying productions, however, is not sufficient in itself. Give a computer a problem and a set of productions, and what happens? Nothing! As Newell & Simon (1972) recognized early on some kind of control mechanism is necessary to activate the productions.

From a theoretical perspective the fewer mechanisms needed the better. With this in mind, Newell & Simon (1972) originally proposed "means-ends" analysis as a universal control mechanism: Given a problem, select (and apply) productions that will reduce the difference between the goal and the current problem state. Mean-ends analysis seemed reasonable and gradually morphed into chaining (of productions). Empirical results later suggested that other mechanisms also are commonly involved in learning and problem solving: Variations on generalization, abstraction, analogy and other mechanisms have been proposed.

A further limitation of learning mechanisms, as used in production systems, is that they impose essential constraints on implementation. One can add or remove individual productions without fundamentally changing the operation of an ITS. Learning mechanisms, however, necessarily come "hard wired". They cannot be added or removed without fundamentally effecting operation of a production system.

Ohlsson (2009) suggested no end to the number of learning mechanisms that might be needed or desired. The impracticability of identifying all potentially relevant mechanisms is one of the reasons that he and Mitrovic introduced Constraint Based Modeling as a means of reducing complexity in ITS development (e.g., Mitrovic & Ohlsson, 2007).

Quite independently, Polya's (1960) early analyses of mathematical problem solving further suggest that learning mechanisms are, in fact, domain dependent. Polya identified a number of domain specific "heuristics" like the pattern of "two-loci" or "similar figures". Such heuristics are formally equivalent to learning mechanisms, but are more similar in nature to higher order rules in SLT (cf. Ehrenpreis & Scandura, 1974; Wulfeck & Scandura, Chapter 14 in Scandura, 1977). Higher order SLT rules are domain dependent and play a direct role in how new SLT rules are acquired and used.

NOTE: While influential in mathematics education, Polya's (1960) work is not widely known in TICL circles.

SLT Solutions.– SLT takes this analysis further. Existing SLT theory offers a detailed road map going forward, a road map that builds directly on current AuthorIT and TutorIT technologies.

Consider the second or third major advances mentioned earlier:

(2) The ability to systematically identify the higher as well as lower order SLT rules required for success in any given domain, no matter how complex.

(3) The ability to formulate SLT's Universal Control Mechanism (UCM) in a way that is completely independent of the rules and higher order rules necessary for success in any given domain.

I summarize each of these advances and their importance below. Then, I describe how AuthorIT and TutorIT can be extended to support each advance.

Structural (cognitive domain) Analysis (SA) of Complex Domains.– SA takes a fundamentally different approach to the problem. The focus here is on identifying both the higher and lower order SLT rules that must be learned for success. Unlike productions (condition-action pairs), SLT rules are not assumed to be in human minds – nor are higher order rules viewed as hard wired mechanisms. Rather, higher as well as lower order SLT rules (like relational models) are both operationally defined in terms of observable behavior with respect to criterion tasks.

All SLT rules represent what must be learned for success. They provide an explicit basis for both diagnosis and remediation.

Historically, Structural (cognitive domain) Analysis (SA) has been used to systematically identify higher as well as lower order SLT rules. As detailed above, the use of ASTs to replace directed graphs has played an important role enabling automation in the development and delivery of adaptive tutoring systems (cf. Scandura, 1971a, 1973, 1977 where SLT rules are represented as directed graphs or flowcharts and Scandura, 2005, 2007 where SLT rules are represented in terms of ASTs). The process by which higher order SLT rules were constructed, however, was largely subjective.

The way higher order SLT rules were constructed was fine for paper and pencil courseware development (e.g., a workbook by Scandura et al, 1971c) and for experimental research (e.g., 1974a). But it was not sufficiently systematic or precise for automation.

As SA was originally defined, the analyst, typically but not necessarily a subject matter expert or instructional designer, was asked to:

a. define a complex problem domain informally,
b. select a finite set of prototypic problems in that domain,
c. construct an SLT solution rule for solving each prototype problem,
d. construct a higher order SLT rule operating on other SLT rules (for constructing each solution rule),
e. eliminate redundant SLT rules (which can be derived by application of higher order rules to others), and
f. repeat the process as desired, each time resulting in a set of SLT rules that were at once simpler and collectively more powerful in generating power.

SA was continued until the SLT rules and higher order rules identified provide sufficient coverage of the domain (cf. Scandura et al 1974 and Wulfeck & Scandura, 1977).

Analysis of various complex domains (e.g., Scandura et al, 1974, Scandura, 1977, Scandura & Scandura, 1980) shows that as SA proceeds two things happen: The individual rules become simpler but the generating power of the rule set as a whole goes up dramatically, thereby expanding coverage in original domain (esp. see Scandura, et al, 1977; Wulfeck & Scandura, 1977).

NOTE: There is no loss of generality because domains can be incrementally expanded without loss by building successively on prior analyses. For details, see Scandura (2007) for the most complete coverage of the basic theory.

Choosing the appropriate level of analysis in Step c was originally ad hoc. This difficulty was solved as above by the introduction of ASTs. Each individual SLT rule can now be refined successively in whatever degree of precision may be necessary or desirable. .

Step d of constructing higher order rules, however, was still a bottle neck – too subjective for full automation. The key to solution was the following missing link between steps c and d:

> Convert each SLT solution rule in Step c into a higher order problem.

Once a higher order problem has been constructed, higher order SLT rules can be constructed in exactly the same way as all other SLT rules.

Given any problem domain, no matter how complex, the goal of Structural Analysis is to identify a finite set of higher and lower order SLT rules – rules that collectively make it possible to solve a sufficiently broad range of problems in the domain. Unlike production systems, where the focus is on identifying ingredients that might be in human brains, the focus in Structural Analysis is on identifying what must be learned for success.

Consider the following example of SA applied to a Number Series domain (adapted from Example 3 in Scandura (2007)). I have selected this example because it illustrates not only higher order SLT rules that generate new SLT solution rules but also how higher order selection rules come into play. (See Appendix B for other examples.).

Number Series Domain – consisting of sums of whole numbers from 1 up. (Step a above)

1. SME Selects Prototypic Problems (one of potentially many) (Step b above)

   1 + 3 + 5 – ?sum

2. Construct (multiple) SLT Solution Rules (2A, 2B, 2C) for Prototypic Problem (each rule can be refined where desired as above) (Step c above, but wherein each solution rule may systematically be refined as in Fig. 2 above)

   2A 1 + 3 + 5 →3x3→ 9
   2B 1 + 3 + 5 →3x(1+5)/2→ 9
   2C 1 + 3 + 5 →successive addition→ 9

3. Convert each SLT Rule into a Higher Order Problem (This is a critical new Step in identifying higher order SLT rules) (Construct Goal & Given of Higher Order Problem)

Higher Order Problem 3A:
1 + 3 + 5 →3x3→ 9
1 + 3 + 5 + 7 + … →nxn→ Sum

Higher Order Problem 3B:
1 + 3 + 5 →3(1+5)/2→ 9
a + a+d + … + L=a+(n-1)d →n(a+L)/2→ Sum

Higher Order Problem 3C:
1 + 3 + 5 →1+3+5→ 9
a1 + a2 + a3 + … + an →successive addition→ Sum

4. Alternative SLT Higher Order Rules for Solving Higher Order Problems 3A, 3B, 3C (Step d above)

Higher Order Rule 3A: →replace 3 terms by n→
Higher Order Rule 3B: →replace 1 by a, 5 by l, 3 terms by n→
Higher Order Rule 3C: →replace each term by a variable, three terms by n→

The process of SA can be repeated (indefinitely). Step e (above) makes it possible (optionally) to eliminate redundant SLT rules – e.g., rules like 4x4, 5x5, …, 50x50 can be derived by applying higher order rule 3A, for example, to 3x3. Higher order rules make it possible to derive any number of new SLT rules from basic rules.

Notice that each alternative higher order SLT rule has a different domain of applicability. Higher order rule 3A is very efficient but only works with arithmetic series beginning with 1 and having a common difference of 2 – for example, 1 + 3 + 5 + … + 99 →50x50→ 2500. Rule 3B is reasonably efficient and works with all arithmetic series. Rule 3C is relatively inefficient (especially with long series) but works with all number series, arithmetic or otherwise.

(NOTE: For early empirical research on the subject see Scandura, Woodward & Lee 1967; Scandura 1967.)

In effect, three higher order rules are applicable rules in this example. At this stage of SA, an analyst may eliminate redundant rules (as in Step e above). Alternatively, deciding which SLT rule to use is essentially what one must do in many design problems. The acquisition of multiple ways of solving any given problem and of knowing which to select when is a key characteristic of expertise.[6]

In our example, the selection process represents a still higher order problem (so SA is repeated as in the original Step f). The given in the higher order problem consists of the three alternative rules. The goal is to select exactly 1. One higher order SLT selection rule that works can be summarized as:

Case Type-of- Number Series:
a) Starts with 1 with a common difference of 2 → select rule N2
b) Common difference → select rule N(A+L)/2
c) Else →select successive addition

A more general but error-prone selection rule is to simply choose the simplest rule. Domain of applicability was largely ignored in early research. Defining the domain structures associated with higher order SLT rules is essential. Automatically perceived structures play a decisive role in determining which rules to use under what circumstances.

THEORETICAL NOTE FOR THOSE INTERESTED IN TRAINING EXPERTISE: For those who have read my recent monograph (Scandura, 2007) I would like to add one general remark: In that monograph I introduced the notion of higher order SLT automation rules as the mechanism by which more efficient (automated) rules are derived from other rules. Irrespective of how they are learned I suspect that most expertise is gained via the gradual acquisition of efficient, increasingly specialized solution rules. Apparently effortless

[6] The above is a form of what is commonly referred to as knowledge engineering. The main difference is that Structural (domain) Analysis is far more systematic with partially automated tools to support the process.

expert behavior results when previously learned, more efficient SLT rules are selected (via higher order selection rules) for use in more and more situations. In accordance with SLT's Universal Control Mechanism (UCM), these more efficient rules are selected by applying higher order (selection) rules as in all other behavior. The result is increasingly efficient, apparently automated behavior.

The Need for Learning (often called Control) Mechanisms.– All knowledge in SLT is strictly relative: What a person knows is defined by that person's behavior relative to what must be learned for success. This relativistic view of knowledge holds whether the knowledge in question is of a higher or lower order. Whereas lower order SLT rules correspond to productions in expert systems, higher order SLT rules correspond to learning mechanisms.

The question then is what controls the use of SLT rules? History makes it clear that neither means-ends analysis, chaining, nor any other expert system mechanism is sufficient. Furthermore, experience with Structural Analysis makes two things clear

a) All mechanisms that have been proposed MAY play a role in problem solving and

b) Variations on all such rules can systematically be derived via Structural Analysis.

Any automated system capable of solving problems must include some kind of control mechanism. The system must know what SLT rule to use and when. I first proposed Goal Switching for this purpose in an invited talk where I first introduced SLT at AERA in 1970 (published in Scandura, 1971a). Unlike chaining and the like, SLT's goal switching was originally modeled on a very easy to state but very hard to implement truism: *Given a problem for which no solution is immediately available, the problem solver must necessarily first derive a procedure for solving the problem*. Indeed, this truism was so general, and so commonsensical that it took considerable convincing to get supporting experimental research on UCM published in the traditionally very rigorous Journal of Experimental Psychology (Scandura, 1974a).

Goal Switching obviously differed from Newell & Simon's (1972) means-ends analysis. Indeed, Newell served as a reviewer and proposed rejecting several of my articles during this time period, including to one above in the Journal of Experimental Psychology (Scandura 1974a) and another in Artificial Intelligence (Scandura et al, 1974). Fortunately, my counter arguments and other reviews led to their eventual publication.

In fact, however, a major limitation of Goal Switching had nothing to do with validity or relevance. A series of formal experiments (Scandura, 1967), as well as more informal pilot research with subjects as young as 4 years old, demonstrated its (near) universal availability to all learners. The difficulty was in attempts to formally implement Goal Switching in a way that was completely independent of ANY higher order rule (cf. Wulfeck & Scandura, 1977). This was finally accomplished with formalization of SLT's Universal Control Mechanism (UCM) in the early 2000s (see Scandura, 2007, U.S. Patent, 6,275,976). Again, I won't repeat here what is already in print (see Scandura, 2007, for specifics).

In retrospect, one can see why expert systems run into trouble. One reason is that knowledge engineering turned out to be very hard, slow and expensive and that experts couldn't always articulate what they were doing. We have seen above how Structural Analysis, while it certainly does trivialize the problem, at least makes it more tractable. More directly relevant in the present context, the original hope was that there were only a small number of basic learning mechanisms – preferably one. Alas, "means-ends analysis" as originally proposed by Newell & Simon (1972) turned out not to be that mechanism.

SLT's Universal Control Mechanism (UCM), on the other hand, serves this role in unique fashion (Scandura, 2007; cf. Scandura, 1971a, 1973, 1974a,b). UCM is completely independent of SLT rules and higher order rules. More important, and unlike means-ends analysis, chaining and other mechanisms proposed in the expert system world, UCM serves as a common denominator completely independent of any particular problem domain.

An overview of UCM follows (for details see Scandura 2007.):

– Check available rules to see which SLT rules have structures that match the given problem
– Unless exactly one SLT Rule matches, control goes to a deeper level looking for rules whose ranges contain structures that match the given problem (a recursive process)
– Once exactly one SLT rule is found, that rule is applied & a new rule generated
– Control reverts to the previous level & the process continues with checking at the previous level of embedding
– Eventually, the process halts because the problem is solved or processing capacity is exceeded (alternatively a predetermined recursion limit may be set in automated systems)

Measuring knowledge relative to behavior in one form or another is not new. However, being able to explain and predict the behavior of individuals in specific instances distinguishes SLT. This is true even more so where a problem solver does not already know a solution procedure – but must derive one. UCM plays an essential role in the latter process.

NOTE: A historical analogy to Relativity Theory is interesting in this regard. Without assigning more significance than warranted, introduction of UCM in SLT plays a role analogous to constancy of the speed of light in Relativity Theory.

Measuring speed of an object relative to an observer was not especially new or interesting. Add in the constant speed of light, however, and the situation changes. As Einstein showed in 1905 funny things happen when one accounts for the time light takes to reach an observer.

Knowledge is strictly relative in a similar sense. What counts as knowledge is not absolute but necessarily relative to observable behavior.

Behavior with respect to complex domains may be explained via finite sets of higher and lower order SLT rules. But, these SLT rules depend on analyst.

UCM is what holds things together. Together with the SLT rules and higher order rules associated with any given domain, UCM allows explicit predications regarding problem solving behavior in specific instances (Scandura, 1974a).

VI. Needed AuthorIT and TutorIT Extensions

To date, AuthorIT/TutorIT tutorials have only been used to develop tutorials for well-defined math skills. TutorIT does, however, support "chaining" although this technologies has not been put to serious use. The only example to date involves a simple railroad crossing, where TutorIT is fed two simple rules: a) one for turning a signal red or green depending on the location of a train (near or out of a crossing) and b) one for raising or lowering a railroad crossing gate depending on the color of the signal (red-down and green-up). TutorIT is not explicitly told that the gate must go down when the train approaches the crossing and up when it is not.

TutorIT is able to generate correct answers by chaining known rules, where the output of one serves as input to the next as required to generate the correct answer. The answers TutorIT generates are used in turn to evaluate learner responses. Chaining is a small step forward and akin to what is done in contemporary ITS systems.

As outlined above (and detailed in Scandura, 2007; Scandura et al, 2009), however, SLT goes much further. Two objectives are on the near term agenda.

1. In order to teach higher order SLT rules we must be able to systematically identify and precisely represent them. The behavioral equivalent of all other learning mechanisms that have been proposed or used in Intelligent Tutoring Systems (ITS). Or expert systems generally, can be represented as higher order SLT rules.[7] These higher order SLT rules are derived directly via Structural (domain) Analysis (SA) from the problem domain itself.
2. In order for TutorIT to generate solutions to ill-defined problems, we must also be able to formalize and implement SLT's Universal Control Mechanism (UCM).

Both AuthorIT and TutorIT will both have to be extended. First, AuthorIT must support the construction of SLT rules that operate on nodes that are themselves SLT rules.

This can already be done using AuthorIT's SoftBuilder component. SoftBuilder is a fully general development system. It supports the construction of any kind of SLT rule. Any SLT rule, whether of a higher or lower order, can be represented as a Flexform. While sufficient in principle, however, it is extremely complex to construct higher order SLT rules. The basic task is hard enough. But, there is no automated support for refining higher order operations (or data) as is currently the case with AutoBuilder.

SA in SLT does provide the necessary rigor. The major work needed is to add support for what are called dynamic structural refinements and corresponding interaction procedural refinements (e.g., see Scandura, 2007, esp. pp. 195-198). As detailed on pages 194-216, Structural Analysis so extended would make it possible to construct arbitrary higher order SLT rules as needed.

The second major improvement requires replacing TutorIT's current chaining mechanism with SLT's Universal Control Mechanism (UCM). Fortunately, the chaining mechanism is a separable module so its replacement and integration should be straight forward. Furthermore, the UCM design has been detailed in a recent patent. The main challenge is to implement, test and refine as necessary to ensure that all work as designed ready for prime time.

I will not than attempt to detail here either the extended form of Structural Analysis or the UCM (Scandura, 2007), and I certainly don't want to imply that this will be a trivial undertaking. The risks are high. For details I encourage you to study my recent monograph (Scandura, 2007, for SA – esp. pp.216-231 and UCM – esp. 216-231).

What is important here is to understand that extension of AuthorIT and TutorIT will do two major things for us:

1. AuthorIT's AutoBuilder component will fully support Structural (domain) Analysis (SA), enabling it to identify and detail higher as well as lower order SLT rules associated with any given domain.
2. TutorIT enhanced with UCM will be able to solve novel problems in domains, even where it is not explicitly given a SLT solution rule.

Given a complex domain, extension of AutoBuilder will more fully support Structural (domain) Analysis (SA). In addition to arbitrary refinement, AutoBuilder will be able to systematically identify finite but sufficient sets of higher as well as lower order SLT rules. Sufficiency means that collectively these SLT rules will provide what the analyst considers to be "adequate coverage" of the given domain. By "adequate coverage" I mean that the rules collectively provide sufficient coverage in the domain – that solutions can be generated for sufficient numbers and varieties of problems in the domain.

Armed with the UCM and a sufficient set of higher (and lower) order SLT rules associated with a problem domain, TutorIT will be able to dynamically derive new solution rules as needed. TutorIT will also be able to provide systematic tutoring on all requisite higher as well as lower order SLT rules.

Given any domain, TutorIT's ability to generate solutions will depend on adequacy of requisite Structural Analysis

[7] The same is true in case based reasoning (CBR), wherein higher order rules involve analogical thinking. From a SLT perspective, CBR involves higher order rules that map solutions (SLT rules) for one kind of task into solutions for analogous ones (e.g., mapping counting up in addition to counting down in subtraction, or repeated addition in multiplication to repeated subtraction in division).

(SA). In this context, it should be emphasized that SA can be applied iteratively. An analyst may build on the results of SA without starting over. SA is a strictly cumulative. The SLT rules deemed sufficient at one point in time may systematically be superseded later on.

Although TutorIT's interface may have to be enhanced, tutoring on higher order SLT rules will take place exactly as any other SLT rule. Knowledge will still be represented hierarchically, and TutorIT decision making will follow the same rules. Critically important from an implementation perspective, theoretical parsimony is matched by the current AuthorIT and TutorIT technologies. It would be fool hardy to underestimate the effort required, but we do not envision major unknowns.

The extended form of TutorIT will select and present problems. The learner will respond, and TutorIT will see if it is correct and provide feedback. If a response is incorrect, TutorIT will provide diagnostic and remediation as detailed above on each of the rules required to solve the problem.

Notice the efficiencies. Suppose the learner is given a complex problem. Instead of having to pinpoint inadequacies in this complex context, it will be sufficient to identify the individual SLT rules and higher order rules necessary for success. Once this has been done, one can treat each individual SLT as before. All SLT rules, higher as well as lower order, have precisely the same formal structure. Hence, diagnosis and remediation can be carried out in modular fashion.

Comparison with ITS.– Given their dominance, comparison with ITS may be helpful to understand the significance of what all this means. AuthorIT can be used to identify what must be learned for success with arbitrary degrees of precision. No longer does one have to worry about individual learner models as such. The author need be concerned only with identifying what higher and lower order SLT rules must be learned for success. This may be done with arbitrary degrees of precision, either initially or in cumulative fashion as experience and development resources dictate. More important perhaps, AuthorIT is not limited in the same way by the complexity of the domain being analyzed. The sheer complexity of some domains makes them inaccessible to traditional ITS methodology. Traditional ITS development requires coming up de novo with: a) a sufficient set of productions, b) assumptions as to what learning mechanisms to use and c) finally data supporting validity of the analysis.

Structural Analysis does not have the same limitations. What one identifies is whatever an expert in the field believes is necessary and sufficient for success in that domain. Certainly, experts may differ as to what they believe should or might be learned. That is not the point. There is nothing to constrain SA to a single point of view. Complications in supporting multiple perspectives include introducing higher order selection rules for deciding which of the alternative solution rules to use under what conditions. In short, anything that can be done with production systems can be done more simply and with TutorIT in conjunction with higher and lower order SLT rules.

Given a representation of what needs to be learned, whether of just lower order as at present or including higher order knowledge as proposed, TutorIT can quickly and easily construct individual learner models, and maintain them dynamically during the course of tutoring. Most important, an extended TutorIT would be able to address diagnosis and remediation on each SLT rule in strictly modular fashion. The result would be orders of magnitude reduction in (pedagogical) decision making complexity. This is simply not possible in a production systems environment.

As above, TutorIT will work even in the face of incomplete analysis. Even a small amount of analysis is better than little or none. Given its complexity, ITS research can only go so far.

None of this means that we should give up on fundamentals. Most TICL research today is limited to general models or frameworks. Some even come with fancy names. I believe we can do more, however, than introduce acronyms in our research.

We need to concentrate more heavily on identifying what needs to be learned. Fifty years of basic and applied research in the field convinces me that the more precisely one understands what needs to be learned the better job one can do of teaching it. This holds whether one is talking about automated tutorials or human teachers. The only difference is that the former can be automated and are more readily subject to incremental improvement.[8]

Comparison with Contemporary TICL Research.– Compare the above also with what is currently done in contemporary tutoring and simulation systems. In such systems, so-called scaffolding is typically indirect, or at best imprecise. We used to call them "Hints". Hints can certainly encourage, indeed require involvement of the learner. If successful, they may also exercise the learner's cognitive abilities.

The problem here is that existing systems of this type have never achieved results comparable to what a skilled human tutor might do. Given increasingly precise representations of what must be learned for success, on the other hand, TutorIT will be capable of providing arbitrarily precise instruction.

Encouraging learners to exercise whatever they may (or may not know) is a good thing. Nonetheless, two points need to be emphasized.

1. There is nothing that forces TutorIT to be as precise as may be possible.

[8] One might think that we already know what needs to be learned in school math. Although analyses in school math tend to be more complete than in other areas, the analyses we have undertaken show that those used for planning textbooks, lessons, CBI programs and even ITS are invariably incomplete. It is not sufficient to simply list the kinds of problems to be solved, to name the particular skills required or even to identify all of the productions that might be involved in solution. Complete analysis requires full systematic analysis of what needs to be learned at all meaningful levels of abstraction. Without full analysis, an automated tutorial will necessarily be incomplete, and cannot reliably guarantee mastery (at least not without including a lot of redundancy).

2. There will inevitably learners for which typical scaffolding is insufficient.

AuthorIT's support for arbitrarily precise representation, extended to include higher order knowledge, will make it possible to reach those who are unable to succeed on their own. By design, the proposed extension of TutorIT would be capable of precise diagnosis and remediation on higher as well as lower order knowledge.

In this regard, I call attention to an early piece of research on math learning (e.g., Roughead & Scandura, 1968) in which we were able to explicitly identify the higher order rules necessary for success in math problems solving (number series). Once identified, we found that students could be taught those higher order rules directly by exposition. Furthermore, it was impossible to tell the difference between those who were taught the higher order rules by exposition and those who discovered them on their own.

I do believe that student's who discover rules on their own and students who are taught those rules secondarily exercise and may learn additional skills. Students who discover higher order rules may in the process also exercise still higher level skills. Conversely, students who learn by exposition gain more experience understanding complex instruction. The essential point is that being able to identify such higher order knowledge (with arbitrary degrees of precision) inevitably makes it easier for more children to learn such higher order skills. Here, <u>we have another example of where computers might eventually take over tasks that they can do better than humans</u>.

One other point deserves emphasis. TutorIT's commitment to deterministic thinking (cf. Scandura, 1971, 2007) requires a significant change in how one goes about evaluating instruction. In particular, it calls into question the usual measures used in controlled experiments, and specifically, the need for controlled experiments focusing on how much is learned. Well designed and suitably refined TutorIT tutorials build on what students already know and automatically adapt to individual needs during the course of (individualized) tutoring. By its very nature, TutorIT requires learners to demonstrate mastery of what is being taught. IF a learner enters with the necessary prerequisites (which can systematically be identified) AND completes such a tutorial that learner will necessarily have demonstrated mastery of what is being taught. This may sound like a tautology, but it is not. Automated instruction that adapts to individual needs, as does TutorIT, requires a different focus. Rather than comparing what or how much various groups of students learn, <u>the critical issues are whether or not a child is motivated to complete a given tutorial, and how long it takes</u>. Similarly, rather than comparing TutorIT with alternative treatments (e.g., classroom learning), <u>one can easily control and compare alternative delivery (i.e., tutoring) modes without confounding content with methodologies</u>.

<u>*Further Extensions*</u>.–Although discussion is beyond the current scope, it is worth noting that these ideas have implications far beyond tutoring systems. As discussed in Scandura (2007) essentially all expert systems are based on deriving implications from sets of productions governed by learning mechanisms of one sort or another. It would be interesting to compare results of expert systems based on productions + mechanisms versus SLT + UCM. Similarly, it would be nice to compare benefits in automatic problem solving. For that matter, it would be interesting to apply the above approach based on KR in SLT and UCM in areas as diverse as robotics and manufacturing

<u>Where do we go from here</u>? Supporting complex domains will not come without a price. Although our current research makes viability clear, the time and effort required with complex domains will almost certainly be greater than with well-defined domains. Mastery in such domains requires the acquisition of higher as well as lower order knowledge. Identifying such knowledge is not always easy. But as early research demonstrates, this can be done (Roughead &Scandura, 1968; Scandura, 1974, 1977; Scandura et al, 1971c; Scandura & Scandura, 1980). Moreover, the process is now far more systematic and it is a task that is long overdue. I leave the position of TICL Chair this year, and am perhaps at the stage of my career where the term “senior advisor” takes on a double meaning.

That said we have already developed a core of TutorIT math skill tutorials covering the basic facts, whole number algorithms and fractions. We plan to add pre-algebra skills in the near term. These tutorials represent only a beginning, but point the way toward a whole new generation of automated (and highly adaptive) tutorials. They also open heretofore-unavailable research opportunities, making it possible to better understand the benefits and limitations of various pedagogies. (As above, measurement should be more in terms of learning efficiencies as opposed to skills being learned.)

More generally, TutorIT technologies have the potential of revolutionizing the way adaptive tutorials are developed, delivered and evaluated. Not only can they be used to develop math skill tutors but highly adaptive tutorials in essentially any area: mathematics, reading, science or otherwise.

While currently supporting development ourselves, we can only go so far alone. Accordingly, I invite those of you who may be interested to join us in the effort. You can help either by making others aware of TutorIT Math tutorials and/or by joining in future development. If interested in developing TutorIT tutorials in your own field of expertise, feel free to contact me at *scandura@scandura.com* .

Together, I believe we can make a real difference. In addition to AuthorIT and TutorIT technologies themselves, we now have in place a unique means of distribution. Anyone can get free TutorIT tutoring time by going to *www.TutorITmath.com*. Furthermore, users can earn more free time by referring others – who will also get free time. It will be exciting to see the results of half a century of research finally making a difference.

REFERENCES

[1] Anderson, J. R., *Rules of the mind*. Hillsdale, NJ: Erlbaum, 1993.

[2] Anderson, J. R., Boyle, C. F., Corbett, A. T. and M.W. Lewis, "Cognitive modeling and intelligent tutoring," *Artificial Intelligence*, 42, pp. 7-49, 1990.

[3] Bloom, B. S., "The 2 Sigma Problem: the Search for Methods of Group Instruction as Effective as On-to-One Tutoring," *Educational Researcher*, 13, 6, pp.4-16.

[4] Dijkstra, S., Schott, F., Seel, N., Tennyson, R. D. Mahwah, NJ, *Instructional Design: Direct Instruction*. Erlbaum, 1997 (Routledge, 1985).

[5] Durnin, J.H. & Scandura, J.M., "An algorithmic approach to assessing behavior potential: Comparison with item forms and hierarchical analysis," *Journal of Educational Psychology*, 65, pp. 262-272, 1973.

[6] Ehrenpreis, W. & Scandura, J.M., "Algorithmic approach to curriculum construction: A field test," *Journal of Educational Psychology*, 66, pp. 491-498, 1974.

[7] Foshay, R. & Preese, F., "Do We Need Authoring Systems? A Commercial Perspective," *Technology, Instruction, Cognition & Learning (TICL),* 2, 3, pp. 249-260, 2005.

[8] Foshay, R. & Preese, F., "Can We Really Halve Development Time: Reaction to Scandura's Commentary," *Technology, Instruction, Cognition & Learning (TICL),* 3, 1-2, pp. 191-194, 2006.

[9] Gagne, R. M., *Conditions of Learning*. NY: Holt, Rinehart & Winston, 1965 (First Edition).

[10] Greeno, J. G. & Scandura, J. M., "All-or-none transfer based on verbally mediated concepts," *Journal of Mathematical Psychology*, 3, pp. 388-411, 1966.

[11] Koedinger, K. R., Anderson, J. R., Hadley, W. H. and M.A. Mark, "Intelligent Tutoring Goes to School in the Big City," *Int. J. Artificial Intelligence in Education*, 8, pp. 30-43, 1997.

[12] Merrill, M. D., *Principles of Instructional Design*. Educational Technology Publications, 1994, 465 p.

[13] Mitrovic, A., Koedinger, K. and B. Martin, "A Comparative Analysis of Cognitive Tutoring and Constraint-Based Modeling," in P. Brusilovsky, A. Corbett, F. de Rosis (eds) *Proc. 9th Int. Conf. User Modeling*, Springer-Verlag, LNAI 2702, 2003, pp. 313-322.

[14] Murray, T., "An Overview Of Intelligent Tutoring System Authoring Tools: Updated Analysis of the State of the Art," in T. Murray, S. Blessing & S. Ainsworth (Eds.), *Authoring tools for advanced technology learning environments*, Chapter 17, The Netherlands: Kluwer, 2003.

[15] Newell, A. & Simon, H.A., *Problem Solving*. Englewood Cliffs, NJ: Prentice Hall, 1972.

[16] Ohlsson, S. & Mitrovic, A., "Fidelity and Efficiency of Knowledge Representations for Intelligent Tutoring Systems," *Technology, Instruction, Cognition & Learning (TICL),* 4, 2-4, pp. 101-132, 2007.

[17] Paquette, G., "Graphical Ontology Modeling Language for Learning Environments," *Technology, Instruction, Cognition & Learning*, 5, 2-4, pp. 133-168, 2007.

[18] Polya, G., *Mathematical discovery (Volume I)*. NY: Wiley, 1962.

[19] Reigeluth, C. M., *Instructional Design Theories and Models.* Mahwah, NJ: Erlbaum, 1983.

[20] Reigeluth, C. M., *Instructional Design Theories in Action.* Mahwah, NJ: 1987.

[21] Ritter, S., "Authoring model-tracing tutors," *Technology, Instruction, Cognition & Learning*, 2, pp. 231-247, 2005.

[22] Roughead, W.G. & Scandura, J.M., " 'What is learned' in mathematical discovery," *Jr. Educational Psychology*, 59, pp. 283-298, 1968.

[23] Scandura, J. M. "The teaching-learning process; an exploratory investigation of exposition and discovery modes of problem solving instruction," *Dissertation Abstracts*, 23, No. 8, 2798, 1963,.

[24] Scandura, J. M., "Abstract card tasks for use in problem solving research," *Journal of Experimental Education*, 33, pp. 145-148, 1964 (a).

[25] Scandura, J. M., "An analysis of exposition and discovery modes of problem solving instruction," *Journal of Experimental Education*, 33, pp. 149-159, 1964 (b).

[26] Scandura, J.M., Woodward, E., & Lee, F., "Rule generality and consistency in mathematics learning," *American Educational Research Journal*, 4, pp. 303-319, 1967.

[27] Scandura, J.M., "Learning verbal and symbolic statements of mathematical rules," *Journal of Educational Psychology*, 58, pp. 356-364, 1967.

[28] Scandura, J. M. & Roughead, W. G., "Conceptual organizers in short-term memory," *Journal of Verbal Learning and Verbal Behavior*, 6, pp. 679-682, 1967.

[29] Scandura, J.M., "Concept dominance in short term memory," *Journal of Verbal Learning and Verbal Behavior*, 6, pp. 461-469, 1967.

[30] Scandura, J.M. & Durnin, J.H., "Extra-scope transfer in learning mathematical rules," *Journal of Educational Psychology*, 59, pp. 350-354, 1968.

[31] Scandura, J.M., "The role of rules in behavior: Toward an operational definition of what (rule) is learned," *Psychological Review*, 77, pp. 516-533, 1970,2

[32] Scandura, J.M., "The role of higher-order rules in problem solving," *Journal of Experimental Psychology*, 120, pp. 984-991, 1974.

[33] Scandura, J. M., "Deterministic Theorizing in Structural Learning: Three levels of empiricism," *Journal of Structural Learning*, 1971 (a).

[34] Scandura, J. M., *Mathematics: Concrete Behavioral Foundations*. NY: Harper & Row, 1971 (b).

[35] Scandura, J. M., Durnin, J.H. & Ehrenpreis, W. N., *Algorithmic approach to Mathematics: Concrete Behavioral Foundations*. NY: Harper & Row, 1971 (c).

[36] Scandura, J. M., *Structural Learning 1: theory and Research*. London//NY: Gordon & Breach Sci. Pub., 1973.

[37] Scandura, J. M., "The role of higher-order rules in problem solving," *Journal of Experimental Psychology*, 120, pp. 984-991, 1974 (a).

[38] Scandura, J.M., Durnin, J.H., & Wulfeck, W.H., "Higher-order rule characterization of heuristics for compass and straight-edge constructions in geometry," *Artificial Intelligence*, 5, pp. 149-183, 1974 (b).

[39] Scandura, J.M., *Problem Solving: a Structural / Process Approach with Instructional Implications*. New York: Academic Press, 1977.

[40] Scandura, J.M. & Scandura, A.B., *Structural Learning and (Piagetian) Concrete Operations*. NY: Praeger Sci. Pub., 1980.

[41] Scandura, J.M., "A cognitive approach to software development: The PRODOC environment and associated methodology," in D. Partridge (Ed.) *Artificial Intelligence and Software Engineering*. Norwood, NJ: Ablex Pub., 1991, Chapter 5, pp. 115-138.

[42] Scandura, J.M., "Automating renewal and conversion of legacy code into ada: A cognitive approach," *IEEE Computer*, pp. 55-61, April 1994.

[43] Scandura, J.M., "Cognitive analysis, design and programming: generalization of the object oriented paradigm," in *Proceedings of the National Ada Conference*, Valley Forge, PA, March 1995.

[44] Scandura, J.M. and Scandura, Alice B., "Improving RAM in Large Software System Development and Maintenance," *Journal of Structural Learning and Intelligent Systems*, 13, pp. 227-294, 1999.

[45] Scandura, J.M., "Structural (Cognitive Task) Analysis: An Integrated Approach to Software Design and Programming," *Journal of Structural Learning and Intelligent Systems (a special monograph)*, 14, 4, pp. 433-458, 2001.

[46] Scandura, J. M., "AuthorIT: Breakthrough in Authoring Adaptive and Configurable Tutoring Systems?" *Technology, Instruction, Cognition & Learning (TICL)*, 2, 3, pp. 185-230, 2005.

[47] Scandura, J.M., "How to Cut Development Costs in Half: Comment on Foshay & Preese," *Technology, Instruction, Cognition & Learning (TICL),* 3, 1-2, pp. 185-190, 2006 (a).

[48] Scandura, J.M., "Reaction to Foshay & Preese Reaction," *Technology, Instruction, Cognition & Learning (TICL),* 3, 1-2, PP. 195-197, 2006 (b).

[49] Scandura, J. M., "AST Infrastructure in Problem Solving Research," *Technology, Instruction, Cognition & Learning (TICL)*, 3, 3-4, pp. 1-13, 2006 (c).

[50] Scandura, J. M., "Learning Objects: Promise versus Reality," *Technology, Instruction, Cognition & Learning (TICL)*, 3, 3-4, pp. 25-31, 2006 (d).

[51] Scandura, J. M., "Converting Conceptualizations into Executables: Commentary on Web-based Adaptive Education and Collaborative Problem Solving," *Technology, Instruction, Cognition & Learning (TICL),* 3, 3-4, pp. 345-354, 2006 (e).

[52] Scandura, J. M., "Knowledge Representation in Structural Learning Theory and Relationships to Adaptive Learning and Tutoring Systems," *Technology, Instruction, Cognition & Learning (TICL),* Vol. 5, pp. 169-271, 2007.

[53] Scandura, J. M., “Introduction to Knowledge Representation, Construction Methods, Associated Theories and Implications for Advanced Tutoring/Learning Systems,” *Technology, Instruction, Cognition & Learning (TICL)*, Vol. 5, pp. 91-97, 2007.

[54] Scandura, J. M., Koedinger, K, Mitrovic, T, Ohlsson, S. & Paquette, G., “Knowledge Representation, Associated Theories and Implications for instructional Systems: Dialog on Deep Infrastructures,” *Technology, Instruction, Cognition & Learning (TICL)*, 6. pp.125-149, 2009.

Software Engineering Publications:

[55] Scandura, J.M., “Cognitive technology and the PRODOC re/NuSys Workbench™: a technical overview,” *Journal of Structural Learning and Intelligent Systems*, 11, pp. 89-126, 1992.

[56] Scandura, J.M., “A cognitive approach to software development: The PRODOC environment and associated methodology,” in D. Partridge (Ed.), *Artificial Intelligence and Software Engineering*. Norwood, NJ: Ablex Pub., 1991, Chapter 5, pp. 115-138.

[57] Scandura, J.M., “Automating renewal and conversion of legacy code. Software Engineering Strategies,” NY: Auerbach Publications, 1994, March/April, pp. 31-43. (Reprinted in *Handbook of Systems Management, Development and Support*, Auerbach, 1995. Similar version in Scuola estiva: Engineering of existing software. Bari, Italy: Dipartimento di Informatica Universita degli Studi di Bari, 1994, pp. 179-192.)

[58] Scandura, J.M., “Automating renewal and conversion of legacy code into ada: A cognitive approach,” *IEEE Computer*, pp. 5-61, April 1994.

[59] Scandura, J.M., “Cognitive analysis, design and programming: next generation OO paradigm,” *Journal of Structural Learning and Intelligent Systems*, 13, 1, pp. 25-52, 1997.

[60] Scandura, J.M., “A cognitive approach to reengineering,” *Crosstalk, The Journal of Defense Software Engineering*, 10, 6, pp. 26-31, June 1997.

[61] Scandura, J.M. and Scandura, Alice B., “Improving RAM in Large Software System Development and Maintenance,” *Journal of Structural Learning and Intelligent Systems*, 13, pp. 227-294, 1999.

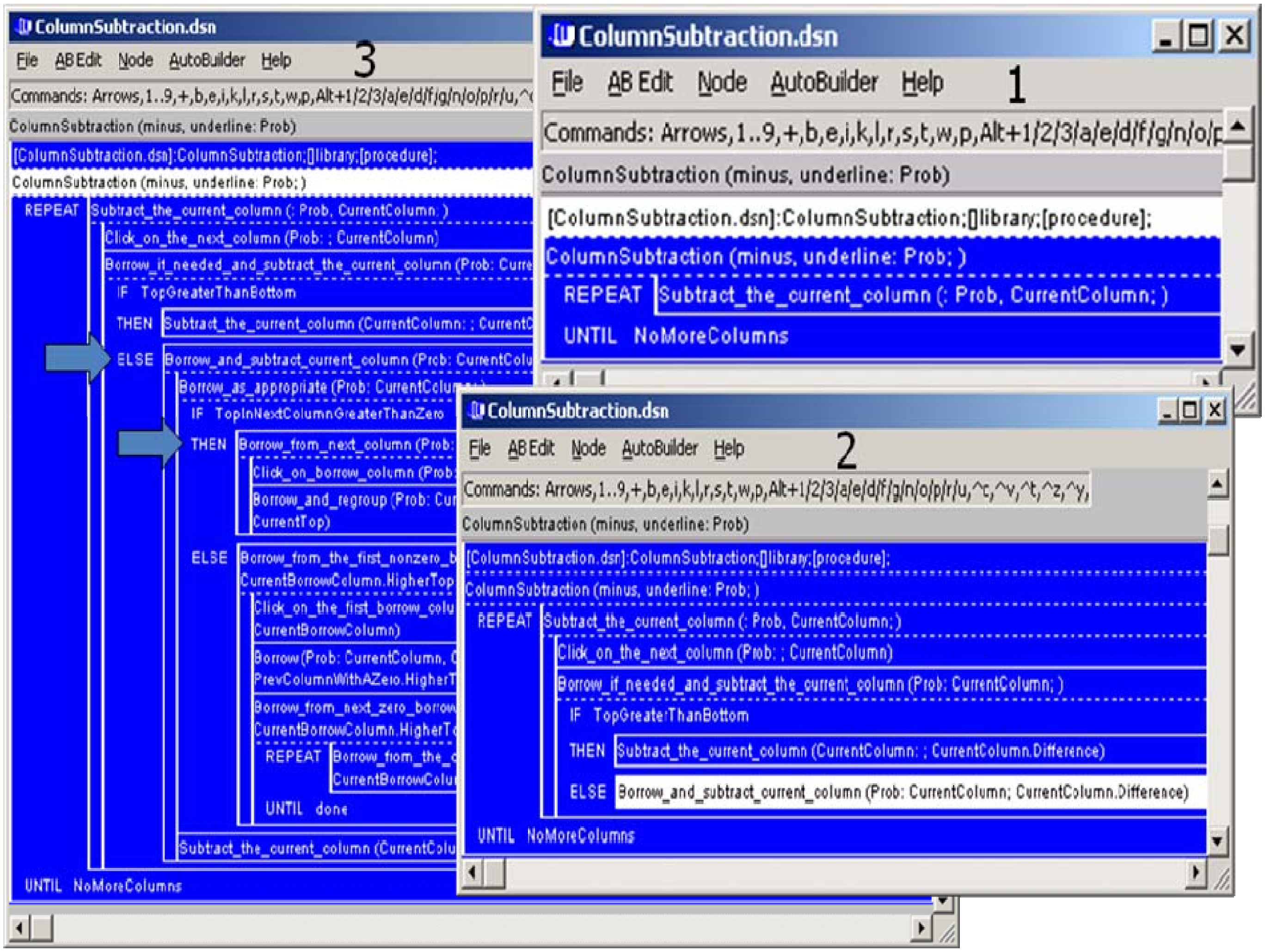
[62] Scandura, J.M., “Structural (Cognitive Task) Analysis: An Integrated Approach to Software Design and Programming,” *Journal of Structural Learning and Intelligent Systems (a special monograph)*, 14, 4, pp. 433-458, 2001.

Key Patent & Pending:

[63] Scandura, J.M., U.S. Patent No. 6,275,976. *Automated Methods for Building and Maintaining Software Based on Intuitive (Cognitive) and Efficient Methods for Verifying that Systems are Internally Consistent and Correct Relative to their Specifications*. August 14, 2001.

[64] Scandura, J.M., Method for Building Highly Adaptive Instruction. US Patent pending.

APPENDIX A

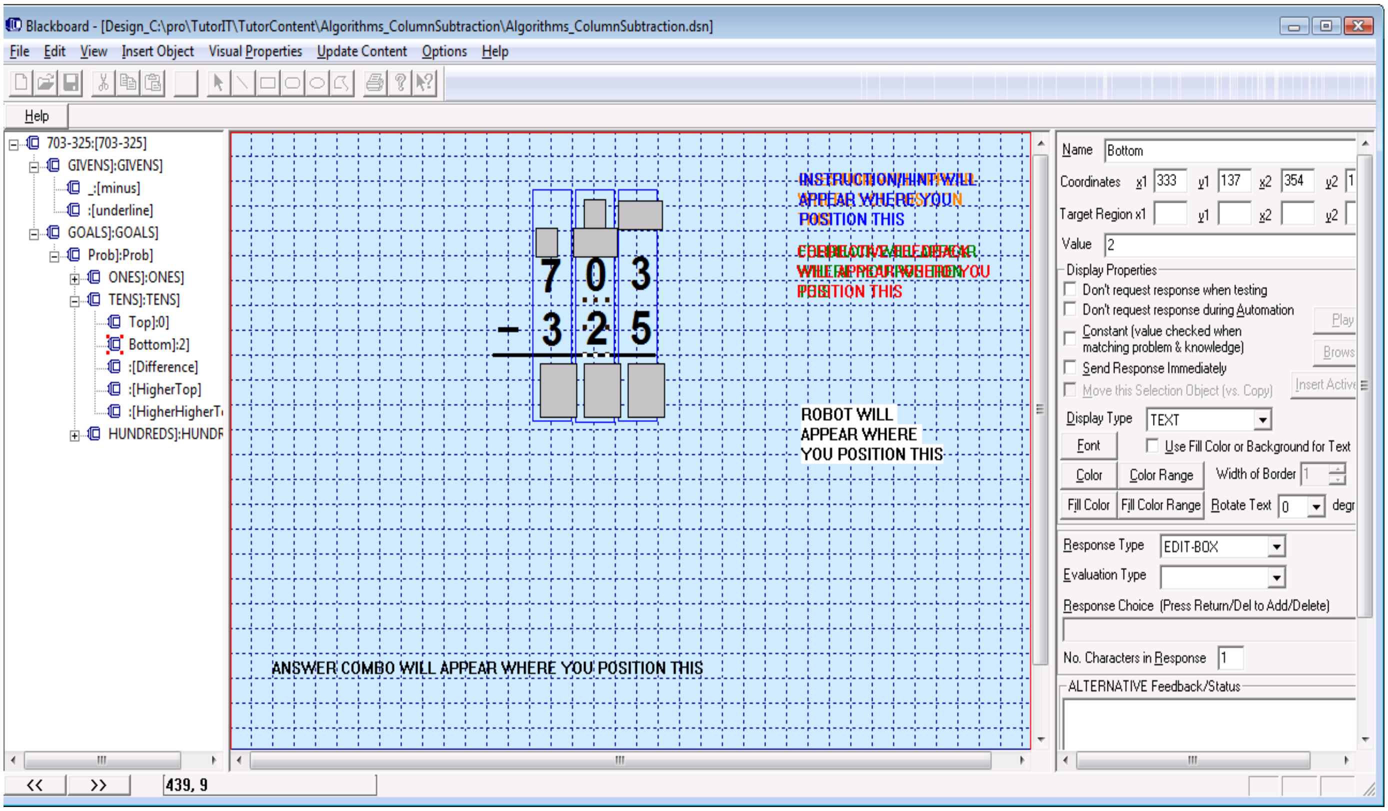


```
Commands: Arrows,1..9,+,b,e,i,k,l,r,s,t,w,p,Alt+1/2/3/a/e/d/f/g/n/o/p/r/u,^c
ColumnSubtraction (minus, underline: Prob)
[ColumnSubtraction.dsn]:ColumnSubtraction;[]library;[procedure];
ColumnSubtraction (minus, underline: Prob; )
REPEAT Subtract_the_current_column (: Prob, CurrentColumn; )
          Click_on_the_next_column (Prob: ; CurrentColumn)
          Borrow_if_needed_and_subtract_the_current_column (Prob: Curre
            IF   TopGreaterThanBottom
            THEN Subtract_the_current_column (CurrentColumn: ; CurrentC
            ELSE Borrow_and_subtract_current_column (Prob: CurrentColu
                   Borrow_as_appropriate (Prob: CurrentColum
                     IF   TopInNextColumnGreaterThanZero
                     THEN Borrow_from_next_column (Prob:
                            Click_on_borrow_column (Prob:
                            Borrow_and_regroup (Prob: Cur
                            CurrentTop)
                     ELSE Borrow_from_the_first_nonzero_b
                          CurrentBorrowColumn.HigherTop
                            Click_on_the_first_borrow_colu
                            CurrentBorrowColumn)
                            Borrow(Prob: CurrentColumn, C
                            PrevColumnWithAZero.HigherT
                            Borrow_from_next_zero_borrow
                            CurrentBorrowColumn.HigherTo
                              REPEAT Borrow_from_the_c
                                     CurrentBorrowColu
                              UNTIL  done
                   Subtract_the_current_column (CurrentColu
UNTIL  NoMoreColumns
```

```
Commands: Arrows,1..9,+,b,e,i,k,l,r,s,t,w,p,Alt+1/2/3/a/e/d/f/g/n/o/p
ColumnSubtraction (minus, underline: Prob)
[ColumnSubtraction.dsn]:ColumnSubtraction;[]library;[procedure];
ColumnSubtraction (minus, underline: Prob; )
REPEAT Subtract_the_current_column (: Prob, CurrentColumn; )
UNTIL  NoMoreColumns
```

```
Commands: Arrows,1..9,+,b,e,i,k,l,r,s,t,w,p,Alt+1/2/3/a/e/d/f/g/n/o/p/r/u,^c,^v,^t,^z,^y,
ColumnSubtraction (minus, underline: Prob)
[ColumnSubtraction.dsn]:ColumnSubtraction;[]library;[procedure];
ColumnSubtraction (minus, underline: Prob; )
REPEAT Subtract_the_current_column (: Prob, CurrentColumn; )
          Click_on_the_next_column (Prob: ; CurrentColumn)
          Borrow_if_needed_and_subtract_the_current_column (Prob: CurrentColumn; )
            IF   TopGreaterThanBottom
            THEN Subtract_the_current_column (CurrentColumn: ; CurrentColumn.Difference)
            ELSE Borrow_and_subtract_current_column (Prob: CurrentColumn; CurrentColumn.Difference)
UNTIL  NoMoreColumns
```

Fig. 2. Successive levels of procedural refinement in Column Subtraction. This Flexform shows all levels refinement, from the highest levels of abstraction to the point where terminal nodes correspond to presumed prerequisites. In column subtraction these prerequisites include basic subtraction facts, ability to compare numbers as to size, etc. Students are tested on entry to ensure that they have mastered these prerequisites.



Blackboard - [Design_C:\pro\TutorIT\TutorContent\Algorithms_ColumnSubtraction\Algorithms_ColumnSubtraction.dsn]

File Edit View Insert Object Visual Properties Update Content Options Help
Help
<< >>
439, 9

```
703-325:[703-325]
  GIVENS]:GIVENS]
    _:[minus]
    :[underline]
  GOALS]:GOALS]
    Prob]:Prob]
      ONES]:ONES]
      TENS]:TENS]
        Top]:0]
        Bottom]:2]
        :[Difference]
        :[HigherTop]
        :[HigherHigherT
      HUNDREDS]:HUNDR
```

  7 0 3
− 3 2 5

INSTRUCTION/HINT WILL APPEAR WHERE YOU POSITION THIS

CORRECTIVE FEEDBACK WILL APPEAR WHERE YOU POSITION THIS

ROBOT WILL APPEAR WHERE YOU POSITION THIS

ANSWER COMBO WILL APPEAR WHERE YOU POSITION THIS

Name Bottom

Coordinates x1 333 y1 137 x2 354 y2 1

Target Region x1 y1 x2 y2

Value 2

Display Properties

- ☐ Don't request response when testing
- ☐ Don't request response during Automation
- ☐ Constant (value checked when matching problem & knowledge)
- ☐ Send Response Immediately
- ☐ Move this Selection Object (vs. Copy)

Display Type TEXT

☐ Use Fill Color or Background for Text

Width of Border 1

Rotate Text 0 degr

Response Type EDIT-BOX

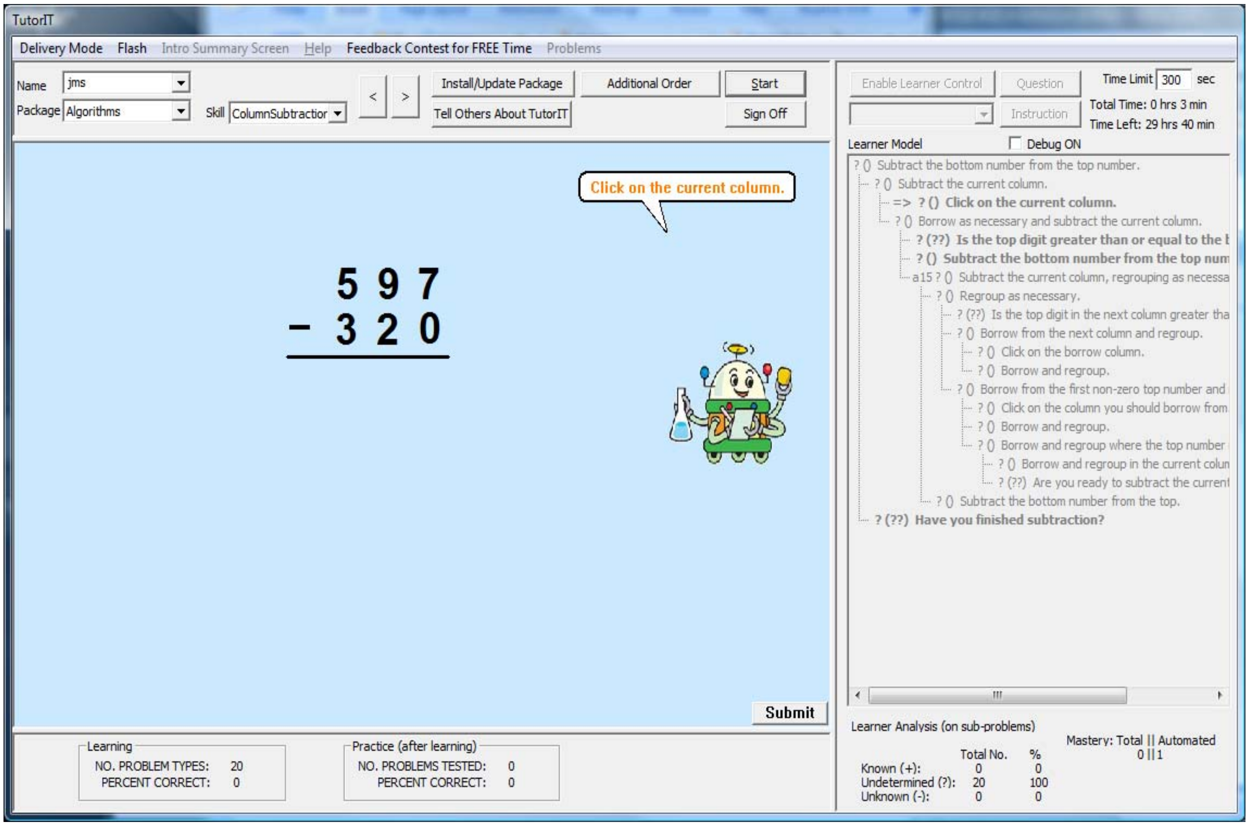
Evaluation Type

Response Choice (Press Return/Del to Add/Delete)

No. Characters in Response 1

ALTERNATIVE Feedback/Status

Fig 3. The left panel in AuthorIT's Blackboard Editor (BB) Editor is used to define individual problems. The center pane is used to layout the interface through which TutorIT and the learner are to interact. It also shows where instruction, questions, positive and corrective feedback are to appear (some appear in the same position, but not at the same time). The right panel is used to assign attributes to individual nodes (elements) in the problem. These attributes include Display types (e.g., Text, Flash, Animation, Sound, Picture, OLE), Response types (Edit Box, Click, Combo Box, Construction) and corresponding Evaluation types (Match_text, Within_region, Structure, Debug).



TutorIT

Delivery Mode Flash Intro Summary Screen Help Feedback Contest for FREE Time Problems
< >
Install/Update Package
Additional Order
Start
Tell Others About TutorIT
Sign Off
Enable Learner Control
Question
Instruction
Submit

Name jms

Package Algorithms

Skill ColumnSubtractior

Time Limit 300 sec

Total Time: 0 hrs 3 min

Time Left: 29 hrs 40 min

Click on the current column.

  5 9 7
− 3 2 0

Learner Model ☐ Debug ON

```
? () Subtract the bottom number from the top number.
  ? () Subtract the current column.
    => ? () Click on the current column.
    ? () Borrow as necessary and subtract the current column.
      ? (??) Is the top digit greater than or equal to the b
      ? () Subtract the bottom number from the top num
      a15 ? () Subtract the current column, regrouping as necessa
        ? () Regroup as necessary.
          ? (??) Is the top digit in the next column greater tha
          ? () Borrow from the next column and regroup.
            ? () Click on the borrow column.
            ? () Borrow and regroup.
          ? () Borrow from the first non-zero top number and
            ? () Click on the column you should borrow from
            ? () Borrow and regroup.
            ? () Borrow and regroup where the top number
              ? () Borrow and regroup in the current colun
              ? (??) Are you ready to subtract the current
        ? () Subtract the bottom number from the top.
  ? (??) Have you finished subtraction?
```

Learning

NO. PROBLEM TYPES: 20

PERCENT CORRECT: 0

Practice (after learning)

NO. PROBLEMS TESTED: 0

PERCENT CORRECT: 0

Learner Analysis (on sub-problems)

	Total No.	%
Known (+):	0	0
Undetermined (?):	20	100
Unknown (-):	0	0

Mastery: Total || Automated
0 || 1

Fig. 5. Learner Model for a student just beginning TutorIT Column Subtraction.
The initial status on each node is set to "?" because TutorIT the student has had some exposure to column subtraction.

APPENDIX B

Steps 1-6 are illustrated in the following three examples, with emphasis on steps 3 & 4 – which are essential in dealing with higher order knowledge. These examples illustrate how higher order SLT rules are derived using three simple domains: Measure Conversion, Proofs in Trigonometry and Number Series. Only top level SLT rules are shown below: Once a top level SLT rule has been constructed, refinement proceeds as above with all SLT rules, whether higher or lower order.

Example 1: Measure Conversion

1.SME Selects Prototypic Problems

3 yd – ?in

2 gallon– ?pints

2. Construct Solution Rules for Prototypic Problems

yd →36_times→ in

gallons →8_times→ pints

3. Convert SLT (solution) Rule to Higher Order Problem

(Construct Goal & Given of Higher Order Problem)

Givens: *yd →n_1_times→ xxx*

xxx →n_2_times→ in

Goal: *blug →n_times→ clug*

4. Construct SLT Higher Order Rule Composition Problem

(Domain/Range Structure of H. O. Rule is Un-initialized Version of Higher Order Problem)

DOMAIN*: *blug [n_times] xxx*

xxx [n_times]) clug

RANGE: *blug (n_times) clug*

Construct Procedure for Higher Order SLT (Composition) Rule

PROCEDURE: ***compose rules so output of first matches input to second***

Example 2: Proving Trigonometry Identities

1. SME Selects Prototypic Problem (one of many)

sin^{2}A + cos^{2}A = 1 – ?proof

2. Construct Solution Rules for Prototypic Problems

sin^{2}A + cos^{2}A = 1 → divide $a^2 + b^2 = c^2$ by c, substitute sin, cos definitions→
Proof is resulting steps

3. Convert SLT Rule to Higher Order Problem (Replace Semantic-specific Nodes in Solution Rule with Abstractions & Select Rule(s)

Given

Goal

sin^{2}A + cos^{2}A = 1 → divide $a^2 + b^2 = c^2$ by c, substitute sin, cos definitions→
Proof is resulting steps

Trig Identity → divide $a^2 + b^2 = c^2$ by side, substitute trig. fn. Definitions→
Proof is resulting steps

4. Construct H.O. SLT Generalization Rule

→Replace Specific Values (e.g., c, sin) with Generalizations→
(e.g., c→side; sin→trig_fns)

Higher order rules make it possible to derive any number of new SLT rules from basic rules. A wide variety of conversion problems, for example, can be solved by combining a small number of basic volume, weight, currency, etc. equivalents. Repeating the process (step f in SA) increases the generative power of the SLT rules and higher order rules associated with the ill-defined domain. Analysis of several complex domains (e.g., Scandura et al, 1974, Scandura, 1977, Scandura & Scandura, 1980) shows that as SA proceeds two things happen: The individual rules become simpler but the generating power of the rule set as a whole goes up dramatically, thereby expanding coverage in original domain (esp. see Scandura, et al, 1977; Wulfeck & Scandura, 1977).

To summarize, the Measure Conversion domain in Example 1 includes any number of (known & unknown) conversion problems, all solvable by **chaining** one known role after another. Example 2 outlines a method (higher order SLT rule) for deriving trigonometric identities as **generalizations** of the Pythagorean theorem (similar to Case Based Reasoning). Example 3 (in the main text) illustrates an ill-defined domain where alternative SLT solution rules are commonly taught (or otherwise learned). As above, this leads to identification of higher order SLT selection rules. It is exactly these kinds of selection rules that must be learned to make sound decisions, whether it be in solving verbal problems in mathematics, or otherwise.

A Revision and Experience using Cognitive Mapping and Knowledge Engineering in Travel Behavior Sciences

Maikel León, Gonzalo Nápoles, María M. García, Rafael Bello, and Koen Vanhoof

***Abstract*—All modern society investigates in the field of Travel Behavior because of the significance for all social and economical process of a country. The problems related with travel behavior are not structured; the Artificial Intelligence techniques have a high interest in its solution, specially related with the knowledge representation and the uncertainty. The use of advanced computer techniques like Knowledge Engineering and Cognitive Mapping is also relevant from diverse points of view. A crucial role is played by the process of modeling and defining what will be taken into account in this kind of problems, for that reason in this paper are described some important ideas of how to understand and extract the mental representation of individuals in the decision making and planning of trips, related to daily travels, because this is useful information that can be used in transport demand prediction, analysis and studies.**

***Index terms*—Travel behavior, knowledge engineering, cognitive mapping, mental representation, daily travel.**

I. Introduction

IN the process of transportation planning, travel demand forecast is one of the most important analysis instruments to evaluate various policy measures aiming at influencing travel supply and demand. In past decades, increasing environmental awareness and the generally accepted policy paradigm of sustainable development made transportation policy measures shift from facilitation to reduction and control [1].

Objectives of such Travel Demand Management (TDM) measures are to alter Travel Behavior without necessarily embarking on large-scale infrastructure expansion projects, to encourage better use of available transport resources and to avoid the negative consequences of continued unrestrained growth in private mobility.

As this policy approach is shifting from rather simple supply-oriented measures to more complex TDM measures, the need to effectively analyze, evaluate and implement a range of policy scenarios is giving rise to the awareness that an improved understanding of individual travel choices and behavior is essential to accomplish reliable and policy responsive forecasts.

Therefore, the advanced travel demand models need to embody a realistic representation and understanding of the travel context and the decision-making process of individuals in order to mimic their sensitivity to a wider range of transport policy measures.

Mental representation is a simplified and subjective reconstruction of the reality. It is for that reason critical to understand how individuals construct these representations to mentally simulate possible decisions and choices under specific expected situational conditions [2]. Because individuals hold their mental representations in working memory, and the capacity of that memory is restricted, individuals will experience restrictions on the amount of information that can be represented.

So, mental representations will in general engage a major overview of reality [3]. The term Cognitive Map refers to the internal mental representation of environmental information. Cognitive mapping is essential for spatial behavior and decision-making whether traveling across a continent or traversing an urban area.

The principal purpose of cognitive mapping is to facilitate individuals to make choices related to the spatial environment. Some transportation researchers have begun to engage with cognitive mapping to a restricted scale, acknowledging that travel and transportation systems are influenced by and they influence spatial cognition [4].

To this point, much of the focus in transportation research has been positioned on how cognitive mapping influences path selection, the routes selected by travelers.

However, the relationship between travel and spatial cognition extends beyond route choice. Cognitive mapping encompasses individuals' knowledge not only of potential travel routes but also of destinations themselves, as well as their proximity, purpose, desirability, and familiarity as such, spatial cognition shapes each person's access to opportunities in the urban environment [5].

Modeling approaches have shifted from trip and tour based models of travel demand to activity based models in which the context of daily travel (i.e. the need to perform activities, household interactions, etc.) is accounted for.

Manuscript received June 1, 2010. Manuscript accepted for publication July 20, 2010.

Maikel León, Gonzalo Nápoles, María M. García and Rafael Bello are with Central University of Las Villas, Santa Clara, Cuba (mle, gnapoles, mmgarcia, rbellop@uclv.edu.cu).

Koen Vanhoof is with Hasselt University, Diepenbeek, Belgium (koen.vanhoof@uhasselt.be).

At the same time, a dramatic increase in computational capacity has enabled modeling techniques to evolve from aggregated approaches to large scale microsimulation of individual travel behavior [4].

In order to transfer and transform the knowledge source from individual minds to some explicit knowledge representation, usually denoted as Knowledge Base (KB), that enables the effective use of the knowledge, it is necessary to explore knowledge acquisition methods in organized approaches, to extract from persons a better understanding of the complex relationships between spatial cognition, travel, and other factors, such as socio-economic status, culture, and individual abilities.

All of this with the intention of helping to guide transportation policymakers, seeking to improve accessibility to important resources such as jobs, healthcare, and other amenities. It is essential to capture true individual decision mechanisms in order to improve behavioral realism of these models [3].

II. Mental Representations, Cognitive Maps and Travel Behavior

At the same time as the literature on theories and measurement of cognitive maps is fixed, the links between cognitive maps and travel behavior is less perceptive. Specifically, research on cognitive mapping and travel has tended to focus primarily, in fact almost exclusively, on the fourth and final part of the traditional travel demand analysis process: route choice. In contrast, the first three steps: trip generation, trip distribution, and in particular, mode choice, have been given far less attention by cognitive mapping researchers [6].

Existing opinion appears to specify that, because factors such as cognitive mapping facility, cognitive map knowledge of possible alternatives, navigation and way finding strategies, and preferences for path selection criteria all are supposed to have a considerable impact on travel choices, there is a rising need to include spatial cognition explicitly in models [7].

Cognitive mapping and travel behavior research has centered on how information on what is known about the location, probable destinations, and viable alternatives for any option affects what is known about the network over which travel must take place. The links between cognitive maps and travel choices are essential to comprehend travel behavior.

The scientific literature on household activity modeling, as a conceptually sound and robust way to forecast travel behavior than traditional travel demand modeling is large and increasing. Activity modeling could be enhanced significantly with better information on how modal experience shapes individuals' cognitive maps (see Fig. 1).

In other words, the cognitive maps of people who mostly walk and use public transit may vary systematically from those who are mostly chauffeured in private vehicles, and from those who usually drive [8].

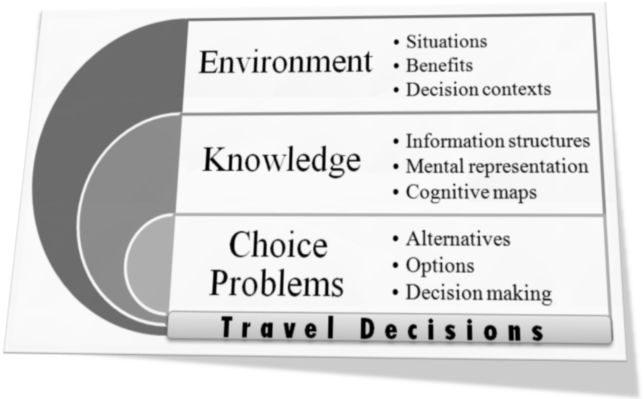


Fig.1. Abstraction levels of mind related to Travel Behavior.

This line of way of thinking is dependable with study on job explore behavior among low salary workers. Those with regular access to private vehicles tend not only to search larger geographic areas work for work, but tend to perceive job opportunities in less spatially constrained ways.

In order to remedy such cognitive barriers to job opportunities experienced by those without regular access to autos, compensatory solutions such as trip planning services, guaranteed ride home services, and overall progresses to transit service could be applied [9].

Another means of compensating for limitations in individuals' cognitive maps could be the dissemination of Intelligent Transportation Systems (ITS). Such systems decrease individuals' overall dependence on their own cognitive maps potentially rising access to recognized destinations. However, ITS would not necessarily influence how prior spatial knowledge informs the initial portions of the travel behavior sequence, trip generation and trip distribution [10].

Persons would still rely on their cognitive maps when choosing to make a trip and selecting a meticulous purpose for that trip. Public transit planning could potentially profit from cognitive mapping study in at least two other ways.

First, the well-documented body of research showing that different people tend to construct and interpret cognitive maps in systematically different ways such as isolated route knowledge as compared to broader configurationally knowledge of a region suggests that the representation of transit networks, routes, transfer points, and schedules might best be consistently represented in redundant ways to be user-friendly to different types of spatial learners [11].

Second, if street and transit networks, while overlapping in space, tend to be constructed completely unconnectedly in the minds of most travelers, this might give details why large shares of personal vehicle drivers never use, or still think using, public transit. While drivers may prefer private vehicle travel over transit, they may never consider using transit, even if a particular transit trip may be competitive in time and cost with an auto trip, if the transit network is, for all intents and purposes, transparent.

However if marketing programs are doing well in encouraging drivers to use transit once or twice, consciousness of transit may cause drivers to change their cognitive maps to include transit as a possibility for a number of trips. Given that high percent of all trips after year 2000 were made in private vehicles, efforts to encourage drivers to occasionally use transit could bear substantial fruit for transit systems anxious to attract more riders.

While cognitive mapping researchers have recognized the connection between travel and spatial learning, little is known yet about how the existing transportation infrastructure itself shapes cognitive maps and, in turn, affects route selection as well as other aspects of travel including trip frequency, trip purpose and destinations, and mode choice.

Nevertheless, the incomplete accessible study suggests that transportation communications and, in particular, way finding on overlapping, up till now distinct, modal networks, sidewalks, bike lanes, local streets and roads, affects the increase of cognitive maps and, in turn, travel behavior [12].

Individual activity travel choices can be considered as actual decision problems, causing the generation of a mental representation or cognitive map of the decision situation and alternative courses of action in the expert's mind. This cognitive map concept is often referred to in theoretical frameworks of travel demand models, especially related to the representation of spatial dimensions.

Actual model applications are scarce, mainly due to problems in measuring the construct and putting it into the model's operation. The development of the mental map concept can benefit the knowledge by individual tracking technologies [4].

At an individual level it is important to realize that the relationship between travel decisions and the spatial characteristics of the environment is established through the individual's perception and cognition of space. Because a person observes space, for instance during travel, the information is added to the individual's mental map.

Among other things, the mental map subsequently shapes the individual's travel decisions, since it reflects what an individual knows and thinks about the environment and its transportation systems (spatial planning).

Although this concept is often referred to in theoretical frameworks of travel demand models, actual model applications are scarce, mainly due to problems in measuring the construct and putting it into the model's operation [13].

III. Knowledge Engineering, Knowledge Acquisition and Knowledge Base

Knowledge Engineering (KE) is defined as the group of principles, methods and tools that allow applying the scientific knowledge and experience to the use of the knowledge and their sources, by means of useful constructions for the human. It faces the problem of building computational systems with dexterity, aspiring first to acquire the knowledge of different sources and, in particular, to conclude the knowledge of the expert ones and then to organize them in an effective implementation.

The KE is the process to design and make operative the Knowledge Based Systems (KBS); it is the topic concerning Artificial Intelligence (AI) acquisition, conceptualization, representation and knowledge application [14].

Traditionally the KE has been related with the software development in which the knowledge and the reasoning play a primordial piece. As discipline, it directs the task of building intelligent systems providing the tools and the methods that support the development of them.

The key point of the development of a KBS is the moment to transfer the knowledge that the expert possesses to a real system (see Fig. 2). In this process they must not only capture the elements that compose the experts' domain, but rather one must also acquire the resolution methodologies that use these [15].

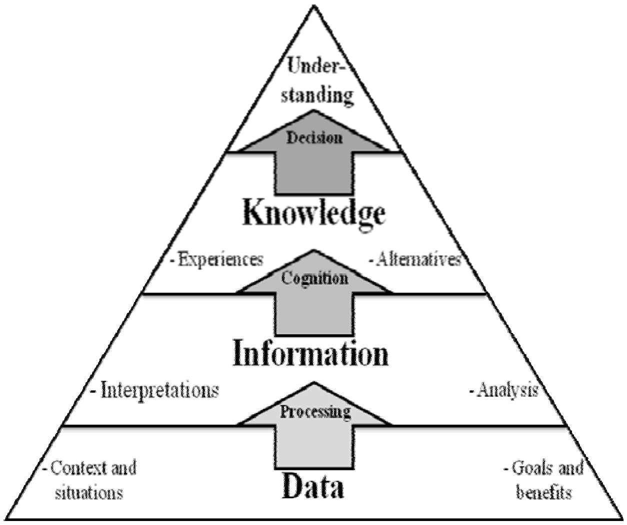


Fig.2. Data, Information and Knowledge Acquisition.

The KE is mainly interested in the fact of "to discover" inside the intellectual universe of the human experts, all that is not written in rules and that they have been able to settle down through many years of work, of lived experiences and of failures.

If the KE can also be defined as the task of to design and build Expert Systems (ES), a knowledge engineer is then the person that carries out all that is necessary to guarantee the success of a development of project of an ES; this includes the knowledge acquisition, the knowledge representation, the prototypes construction and the system construction.

The fundamental problems in the construction of the KBS are [16]:

- Knowledge Acquisition: How to transfer the human knowledge to an effective representation abstract, denominated conceptualization.
- Knowledge Representation: How to represent the knowledge in terms of information structures that a computer can later process.
- Inferences Generation: How to use those information structures to generate useful information in the context of a specific case.

A Knowledge Acquisition (KA) methodology defines and guides the design of KA methods for particular application purposes. Knowledge elicitation denotes the initial steps of KA that identify or isolate and record the relevant expertise using one or multiple knowledge elicitation techniques. A KA method can involve a combination of several knowledge elicitation techniques which is then called knowledge elicitation strategy (Of course these terms are used differently by different authors).

There are several characteristics of KA that need to be considered when applying KA methods [17]. KA is a process of joint model building. A model of expertise is built in cooperation between a domain expert (i.e., the knowledge source) and a knowledge engineer. Appropriate knowledge elicitation techniques are needed to make it plain.

The results of KA depend on the degree to which the knowledge engineer is familiar with the domain of the knowledge to be acquired and its later application. Also, it is noticed that the results of KA depend on the formalism that is used to represent the knowledge. KA is most effective if knowledge representation is epistemologically adequate (i.e., all relevant aspects of expertise can be expressed) and usable (i.e., suits all later usage needs).

These characteristics of KA provide guidance for the design of KA methods. For example, they imply that KA methods must assure that the knowledge engineer becomes familiar with the application domain.

The KA also takes into account the transfer and transformation of the potential of experience in the solution of a problem from several sources to a program. The sources are generally expert human but it can also be empiric data, books, cases of studies, etc.

The required transformation to represent the expert knowledge in a program can be automated or partially automated in several ones [18].

There are different ways of KA:

- The expert interacts with the knowledge engineer to build the KB:
  [Expert] → [Knowledge Engineer] → [KB]
- The expert can interact more directly with the ES through an intelligent publishing program, qualified with sophisticated dialogues and knowledge about the structure of the KBs:
  [Expert] → [Intelligent Program] → [KB]
- The KBs can be built partially by an induction program starting from cases described in books and past experiences:
  [Books] → [Induction Program] → [KB]
- A method of acquisition of the most advanced knowledge is the direct learning from books:
  [Books] → [Data Processing] → [KB]

General requirements exist for the automation of the KA and they should be considered before attempting this automation, such as independence of the domain and direct use of the experts without middlemen, multiple accesses to sources of such knowledge as text, interviews with experts and the experts' observations.

Support to diversity of perspectives including other experts, to diversity of types of knowledge and relationships among the knowledge, to the presentation of knowledge of diverse sources with clarity, in what refers to their derivation, consequences and structural relationships, to apply the knowledge to a variety domain and experience with their applications and to validation studies.

The automated methods for the KA include analogy, learning like apprentice, learning based on cases induction and analysis of decision trees, discovery, learning based on explanations, neural nets, and modification of rules and tools and helps for the modeling and acquisition of knowledge that have been successful applied; they seem to depend on intermediary representations that constitute languages of modeling of problems that help to fill the hole between the experts and the implementations of programs [15].

Diverse causes have taken to the construction of the Automated Knowledge Engineers (AKE), the descent in the cost of the software and hardware for ES, it has favored the development of the same ones. This has increased the demand of ES, greater than the quantity of AKE, and able to support ES.

The movement toward an extensive human activity, as the KE, is contrary to all the industry tendencies, in particular the industry of the software.

The Knowledge Engineer's role, as middleman between the expert and the technology, sometimes is questioned. Not only because it increases the costs but also for their effectiveness, that is to say, it can get lost knowledge or it can influence subjectively on the KB that is making (see Fig. 3).

The automated knowledge acquisition keeps in mind in what measure belong together the description of the application domain that has the expert and the existent description in the KB and how to integrate the new information that the expert offers to the KB.

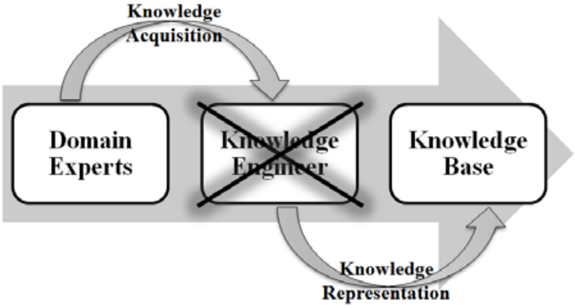


Fig.3. Automated Knowledge Engineer.

The AKE, if it is possible, should be independent of the experts' domain, to be directly applicable for the experts without middleman able to ascend to diverse sources of knowledge, including texts, interviews with the experts, and other features.

Also, it should be able to embrace diverse focuses, even different experts' partially contradictory approaches, and to be able to embrace diverse forms of knowledge representation.

Diverse methods of implementation of AKE exist [17], some of the most significant can be:

- Generation of rules starting from a database whose fields correspond to the attributes or conditions and the last field corresponds to the conclusion. Each article of the base becomes a rule.
- Dialogue with the experts. The AKE should guide the expert, but with certain flexibility.
- Learning for similarity. Given a group of objects which represent examples and opposite of examples of a concept, the AKE generalizes a description that covers the positive examples and not the negatives. The positive examples generalize and the negatives specialize the objects (the concepts can be described as rules).
- Adjustment of numeric parameters of certain parts of the knowledge, as the coefficient of the expressions that conforms the production rules.

Most of the existent methods to acquire the knowledge automatically, work with a fixed representation language, developed by the designer. The training data (examples) for these methods can contain non prospective errors using the knowledge domain to guide the learning. Some methods of automated learning are not strong to select the appropriate generalization of the data, among all the possible ones [19].

IV. Automated Knowledge Engineer for acquiring Individuals Mental Representation about Travel Behavior

While faced through complex choice problem like activity-travel option, persons generate a mental representation that allows them to understand the choice situation at hand and assess alternative courses of action.

Mental representations include significant causal relations from realism as simplifications in people's mind. We have used for the capture of this data, in the knowledge engineering process, an Automated Knowledge Engineer (see Fig. 4), where the user is able to select groups of variables depending of some categories, who characterize what they take into account in a daily travel activity. There are diverse dialogues, trying to guide the user, but not in a strict way or order.

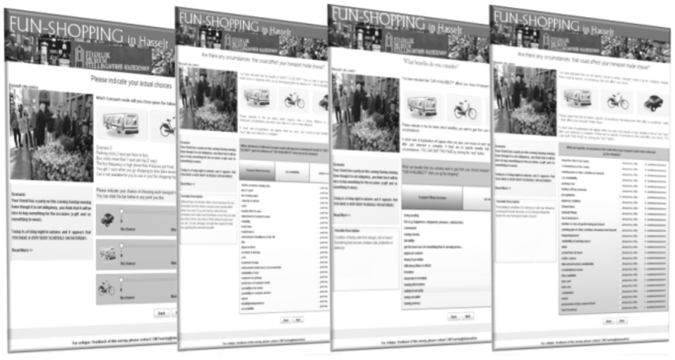
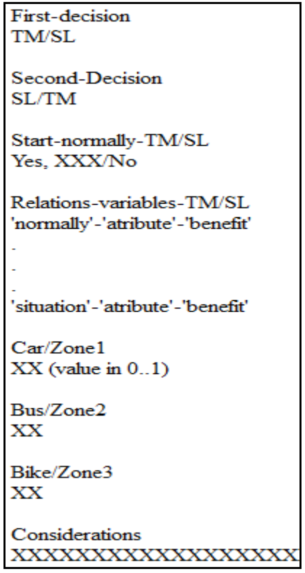

FUN-SHOPPING in Hasselt
Please indicate your actual choices
FUN-SHOPPING in Hasselt
Are there any circumstances that could affect your transport mode choice?
FUN-SHOPPING in Hasselt
What benefits do you consider?
FUN-SHOPPING in Hasselt
Are there any circumstances that could affect your transport mode choice?

Fig.4. Automated Knowledge Engineer.

In the software there are 32 different ways to sail from the beginning to the end, due to the flexibility that must always be in the data capture process, trying to adapt the Interface as much as possible to the user, guarantying then that the given information will be as natural and real as possible, never forcing the user to give an answer or to fill a non-sense page.

For each decision variable selected a matrix with attributes, situational and benefit variables exist, in this way respondents are asked to indicate the causal relations between the variables.

Fig. 5 shows a segment of the definition file that is automatically generated with the flat representation of a cognitive map. This process is totally transparent to the user (that´s way is called Automated Knowledge Engineering).



```
First-decision
TM/SL

Second-Decision
SL/TM

Start-normally-TM/SL
Yes, XXX/No

Relations-variables-TM/SL
'normally'-'atribute'-'benefit'
-
-
-
'situation'-'atribute'-'benefit'

Car/Zone1
XX (value in 0..1)

Bus/Zone2
XX

Bike/Zone3
XX

Considerations
XXXXXXXXXXXXXXXXXXX
```

Fig.5. Definition file segment of the generated KB.

Fig. 6 shows a possible and simple real map of a person after the selection of the variables and the relationship that was considered. Because of individual differences in the content of cognitive maps, different motivations or purposes for travel and different preferences for optimizing or satisfying decision strategies, human travel behavior is difficult to understand or predict.

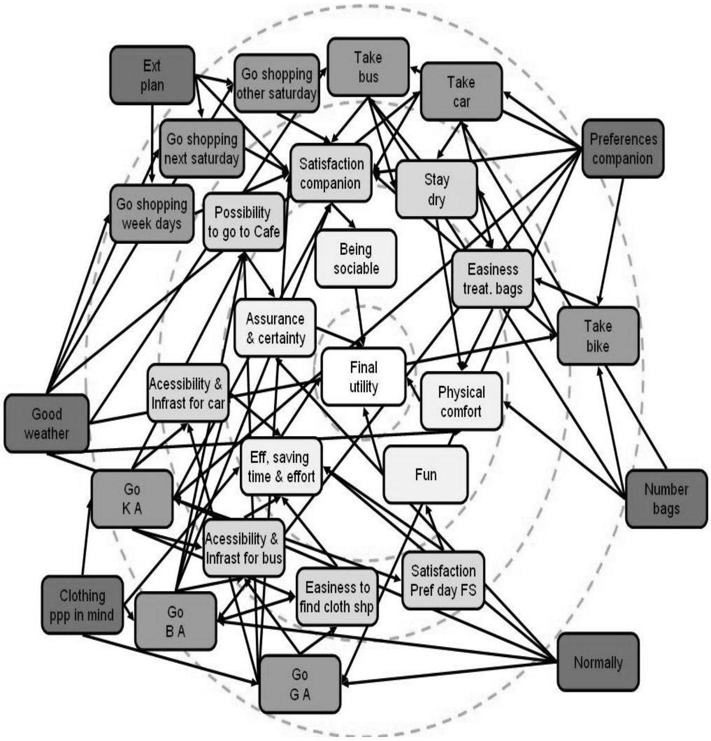


Fig.6. Possible individual cognitive map for a shopping activity.

The problem facing future study is that of combining travel demand with network provide with an understanding of how persons choose on where they prefer to go and how they prefer to get there. Emphasizing cognitive mapping values may give a stage of imminent that has not so far been completely supplied.

In a case study 223 persons were already asked to use the software, and the results are really promising given that the 99% of individuals (see Fig. 7) were able to interact complete along with the Automated Knowledge Engineer, generating their own cognitive map about a shopping activity scenario that was given.

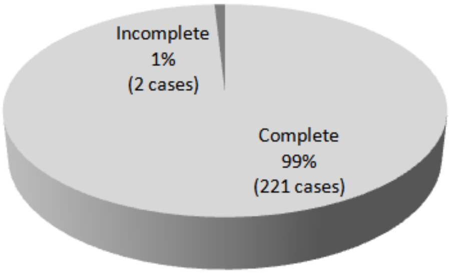


Fig. 7. Percent of complete generated KBs.

V. Conclusions and Future work

We have argument in this paper that cognitive mapping research has the possibility to address the continuing focus on accessibility in transportation studies.

While accessibility has traditionally been conceived as proximity of (or cost of travel between) one location and others, cognitive mapping research shows that physical distances are only one factor shaping how individuals make choices in a spatial context.

Human being differences, including past modal travel experiences, cultural preferences, and spatial abilities, form the cognitive map and, in this manner, influence the cognitive immediacy and openness of latent destinations in a region.

The automated methods used in the Knowledge Engineer in occasions can end up being more competent than the humans to acquire and to refine certain types of knowledge. They can reduce the high cost significantly in human resources that it wraps the construction of Knowledge Based Systems.

It had been taken into account the satisfactory use of the Automated Knowledge Engineering to extract mental representations and as an interesting way of make automatic a cognitive map formalizing.

Considering this, an Automated Knowledge Engineer to acquire Individuals Mental Representation about Travel Behavior was developed, and from a generated Knowledge Base is directly built a Fuzzy Cognitive Maps that characterize the way of thinking of a person, giving us the possibility of simulate the behavior of individuals, to infer and predict future situations that can be considered in the transport planning process.

References

[1] M. Dijst, “Spatial policy and passenger transportation,” *Journal of Housing and the Built Environment*, Vol. 12, pp. 91-111, 1997.

[2] W. Kandasamy, “Applications of Fuzzy Cognitive Maps to Determine the Maximum Utility of a Route,” *Journal of Fuzzy Mathematics*, Vol. 8, pp. 65-77, 2000.

[3] C. Khisty, *Transportation Engineering: an introduction*. Prentice Hall, 1990, 388 p.

[4] S. Krygsman, *Activity and Travel Choice in Multimodal Public Transport Systems*. Utrecht University, 2004.

[5] T. McCray, “Measuring Activity and Action Space/Time: Are Our Methods Keeping Pace with Evolving Behavior Patterns?” in *Integrated land-use and transportation models: behavioral foundations*, Oxford: Elsevier, Chapter 4, 2005, pp. 87-100.

[6] E. Hannes, “Does Space Matter?” in *Travel Mode Scripts in Daily Activity Travel. Environment and Behavior*, 2008.

[7] D. Janssens, “Tracking Down the Effects of Travel Demand Policies”. in *Urbanism on Track. Research in Urbanism Series,* IOS Press, 2008.

[8] P. Taco, “Trip Generation Model: A New Conception Using Remote Sensing and Geographic Information Systems,” in *Photogrammetrie Fernerkundung Geoinformation,* 2000.

[9] P. Jones, *Developments in Dynamic and Activity-Based Approaches to Travel Analysis*. Gower Publishing Company, 1990.

[10] L. Figueiredo, “Intelligent Transportation Systems,” in *IASTED International Conference Applied Simulation and Modeling*. ACTA Press, 2002.

[11] S. Krygsman, *Activity and Travel Choice in Multimodal Public Transport Systems*, Utrecht University, 2004.

[12] T. Garling, *Theoretical Foundations of Travel Choice Modeling*. Pergamon, Elsevier, 1998.

[13] P. Stopher, *Understanding Travel Behavior in an Era of Change*. Pergamon, Oxford University, 1997.

[14] A. Soller, “Knowledge acquisition for adaptive collaborative learning environments,” in *American Association for Artificial Intelligence Fall Symposium*, Cape Cod, MA, AAAI Press, 2000.

[15] P. Cassin, “Ontology Extraction for Educational Knowledge Bases,” in *Spring Symposium on Agent Mediated Knowledge Management,* Stanford University, American Association of Artificial Intelligence, 2003.

[16] B. Woolf, “Knowledge-based Training Systems and the Engineering of Instruction,” in *Macmillan Reference*, New York, Gale Group, 2000, pp. 339-357.

[17] J. Mostow, “Some useful tactics to modify, map and mine data from intelligent tutors,” *Natural Language Engineering*, United Kingdom, Cambridge University Press, Vol. 12, pp. 195-208, 2006.

[18] C. Rosé, “Overcoming the knowledge engineering bottleneck for understanding student language input,” in *International Conference of Artificial Intelligence and Education*, 2003.

[19] A. Jameson, “When actions have consequences: Empirically based decision making for intelligent user Interfaces,” *Knowledge-Based Systems*, Vol. 14, pp. 75-92, 2001.

Creation and Usage of Project Ontology in Development of Software Intensive Systems

Petr Sosnin

Abstract—**The key problem of successful developing of the software intensive system (SIS) is adequate conceptual interactions of designers during the early stages of development. The success of the development can be increased by using of a project ontology, the creation of which is being embedded into the processes of conceptual solving the project tasks and specifying the project solutions. The essence of the conceptual design is a specification of conceptualization. The main suggestion of this paper is a creation of the project ontology in the form of a specialized SIS that supports the conceptual activity of designers. For creation of the project ontology of such type, the instrumental shell was developed. For creation of the project ontology the designers should fill this shell with the adequate information. The basic reasons for evolving the content of the ontology are negative results of testing of the used text units according to the conformity to the ontology. Such shell (in any state of its using) includes the created ontology and its working version (working dictionary) which helps to manage the informational flows, to register the life cycles of the conceptual units and to provide the representativity of their usages.**

Index terms—**Project ontology, system development, software engineering, task solving.**

I. Introduction

NOWADAYS one of the most challenging area of computer applications is "Development of Software Intensive Systems", within the frame of which the collaborative works of developers and other stakeholders are being carried out in corporate networks. The success of such activity in this area, which is being estimated regularly by corporation Standish group [18] for last 16 years, is extremely low (a little more than 30%). Failures can occur in development of the SIS related to any part of the SIS's definition [15]: "A software intensive system is a system where software represents a significant segment in any of the following points: system functionality, system cost, system development risk, development time."

A very important cause of the failures is semantic mistakes in the collective intellectual activity of developers and other persons involved to the development of the SIS. The necessary condition of the developers success is their mutual understanding in collaborative actions based on reasoning over textual information including the statements of task and definitions of project solutions. Developers of the SIS should be supplied with useful and effective techniques for the prevention and correction of semantic mistakes.

At the beginning stage of the SIS development the necessary understanding usually is absent. The adequate understanding is formed only gradually and step by step during the interaction in working groups. Evolution of understanding follows step by step the design of the SIS in the collaborative development environment (CDE) and the current state of understanding includes its positive influences on the management of the development process.

The important role of understanding (personal and mutual) in the development of the SISs is well known. For exploiting of this phenomenon the special techniques for "interactions" with understanding are being created and are used. One type of such technique is a glossary. The specialized version of the glossary is applied, for example, in widely used methodology (and technology) Rational Unified Process (RUP) [14]. Let us notice that in the RUP such artifact is normatively defined, though it does not have collaborative techniques for its informational filling in real time of design. The problems of dynamically extracting, defining, modeling, registering, keeping and visualizing the units of understanding in designing the SISs do not have satisfactory solution.

In this paper for the explicit work with understanding of SIS designers, a specialized system of the project ontology, which is creating as a subsystem embedded into the developing SIS, is proposed. Moreover, it is suggested to create the project ontology as an interactive system of the SIS type. Such system which will be denoted below as $\mathrm{SIS}^{\mathrm{ONT}}$, is implemented on the base of an ontology shell which supports the collaborative extracting and checking of ontology units from statements of project tasks and definitions of project solutions.

The implemented ontology shell is included into the instrumental system WIQA [16] which is aimed to designing the complex system of the SIS type. The WIQA is based on question-answer reasoning and the models of the units' flow, of which the informational source of concepts' usages embedded into the project ontology is extracted.

II. Related Works

A set of typical kinds of ontologies, according to their level of dependence on a particular task or a point of view, includes the top-level ontologies, domain ontologies, tasks ontologies and applied ontologies. All these types of ontologies are defined in [9] and [10] as techniques that are used in different systems.

Manuscript received May 9, 2010. Manuscript accepted for publication August 3, 2010.

Petr Sosnin is with the Computer Department, Ulyanovsk State Technical University, Ulianovsk, 432028, Russia (sosnin@ulstu.ru).

For the SISs, the more adequate type of ontologies is applied – the type that must be expanded usually by means of the other ontologies' types. In accordance with the publication [11], the theory and practice of applied ontologies "will require many more experiences yet to be made".

It is necessary to notice that the project ontology as a subtype of applied ontologies is essentially important for SISs. Project ontologies mainly are aimed at the process of design but after refining they can be embedded into implemented SISs.

The specificity of project ontologies is indicated in a number of publications. In the technical report [5] the main attention is concentrated on "people, process and product" and collaborative understanding in interactions. Investigation of the possibility of the ontology-based project management is discussed in the paper [1].

The usage of the ontology potential in developing the program system and ontological problems of program products are investigated in the paper [4]. This article describes the experience of development of the task ontologies taking into account first of all the role of different kinds of knowledge. The introduction of knowledge into the task ontologies is reflected and discussed in the work [2]. The role of knowledge connected with problem-solving models is presented in the paper [12].

In all mentioned publications there are many useful ideas but the approach to the ontology as to the specialized SIS^{ONT} – for extracting, defining and assembling concepts into the ontology in the process of designing the SIS – is not considered. The Internet search of publications with key words which include such phrases as "project ontology" and "software intensive system", has remained without competitive results coinciding with results suggested in this paper.

Let us remind that the main goal in using the project ontology is to provide the necessary understanding in collaborative design which is impossible without human-computer interaction. Therefore the theory and experience of human-computer interaction as presented in [13] were taken into account in this paper.

III. Specificity of Suggested Ontology

Attempts to view the project ontology from the side of creating the specialized SIS^{ONT} leads to the questions about its architecture, life cycle and used models which must be coordinated with the evolution of the project ontology. Below we answer these questions.

The architecture of any SIS^{ONT} for the definite SIS has a problem-oriented type the materialization which begins its life cycle from the ontology shell with architectural solutions, inherited and kept by the SIS^{ONT} without changing. The principle architecture of the shell (and any SIS^{ONT} also) is presented in Fig. 1.

For any dictionary entry of the ontology there is a corresponding analog in the working dictionary. Such analog is used firstly as a representative set of samples registering the variants of the concept usages extracted from statements of project tasks and definitions of project solutions (or shortly from text units). Samples are being gathered naturally in interactions of designers who are testing (implicitly or explicitly on different working places) the used concepts according to their conformity to the ontology.

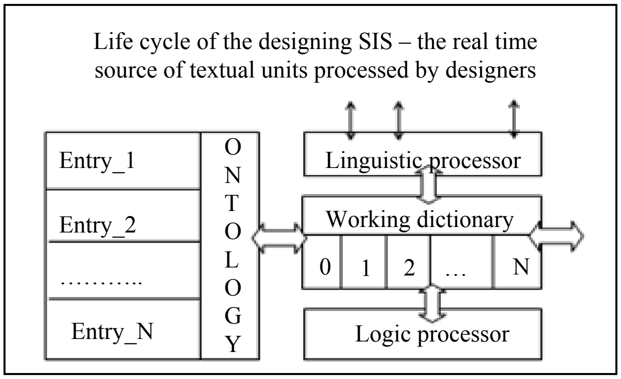


Fig. 1. Architecture of the project ontology.

Filling the ontology by the content is connected with a specialized project task appointed to an administrator of the ontology. The work of the administrator is managed:

- By events each of which is generated when the result of comparison of the used concept with the ontology is not correct;
- In accordance with a sequence of actions supporting the normative state of the project ontology (current levels of adequacy and systematization).

The necessary informational material for the administrator of the ontology is supplied by designers with the help of the predicative analysis. Designers must test and confirm the authenticity of concepts which are used in statements of tasks and definitions of project solutions. For achieving such aim they have to extract firstly the usage of concepts (from the text units) and then to compare them with the ontology. The differences of comparisons (new concepts or additional parts of existing concepts, additional questions which require answers) are used as the informational material for evolving the ontology. Let us notice that any extracted concept usage includes its expression as a simple predicate but not only this (the full expression will be presented below).

Used concepts are the main part of the project ontology which should be expanded by systematizations and axiomatic relations. Techniques of systematizations are embedded into the ontology component while axiomatic relations are being created with the help of the logic processor.

The logic processor is intended to build the axiomatic relations as formulas of the logic of predicates. Such work is being implemented in the frame of the appropriate article (entry) of the working dictionary where the necessary simple predicates are being accumulated. Ontology axioms express materialized units of the SIS and first of all those of them which corresponds to UML-diagrams. Any built axiom is registered in the definite entry (article) of the ontology.

The main architectural view presents the project ontology from the side of its components and informational content which defines the dynamics of the life cycle for the SIS^{ONT}. In a typical case such life cycle is being implemented in the form of the real time work of several dozens of designers who have solved and are solving several thousands of tasks. Models which are used in the ontology life cycle will be presented below.

IV. Linguistic Processor

The life cycle of the SIS^{ONT} is embedded into the life cycle of the designing SIS from which all (named above) text units are being introduced into the linguistic processor. Another possibility is to apply some term-extraction technique, for example, as described in [8].

For testing any text unit, it is transformed into a set of simple sentences and in such transformation the pseudo-physics model of the compound sentence or complex sentence of the other type is applied. In the pseudo-physics model of the sentence all used words are interpreted as objects which take part in the “force interaction” which is visualized on the monitor screen. Formal expressions of pseudo-physics laws are similar to the appropriate laws of the classic physics.

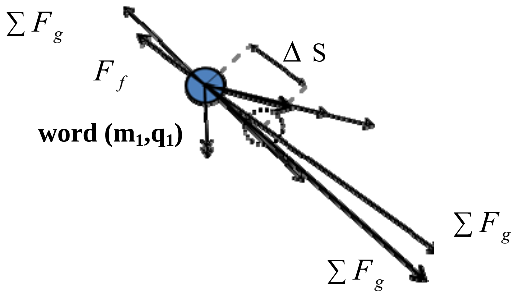


Fig. 2. Interaction of forces.

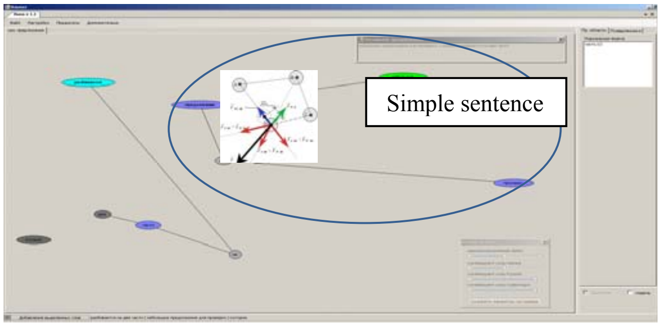


Fig. 3. Extraction of the simple sentence.

In accordance with acting forces (forces of “gravitation” F_g, “electricity” F_q, “elasticity” F_e and “friction” F_f) and attributes appointed to the “word-objects” such objects after moving are being grouped in definite places of the interaction area. The possible picture of the forces interaction for one word of the investigated sentence is shown in Fig. 2.

In the stable state (Fig.3), each group of words-objects will present the extracted simple sentence after finishing dynamic process on the screen.

The screenshot in Fig.3 and other screenshots of this paper are used with labels for the generalized demonstrations of the visual forms and objects with which the designers are working. The language of these screenshots is Russian.

Let us notice that in the assignment of attributes (values of m_i, q_i and others values and parameters) two mechanisms are applied – the automatic morphological analysis and the automated tuning of object parameters. Values are assigned in accordance with the type of the part of speech. The suitable normative values were chosen experimentally. For description of morphological analysis see works [6], [7].

After extraction of simple sentences the designer begins their semantic analysis aimed to testing the correctness of each simple sentence (SS_i). In such work the designer uses the model of SS_i and its relations with surrounding, as presented in Fig. 4. This picture shows the type of SS_i which is used for registering the appointment of the property for object. The other type of model for registering the appointment of relation between two objects has the similar scheme.

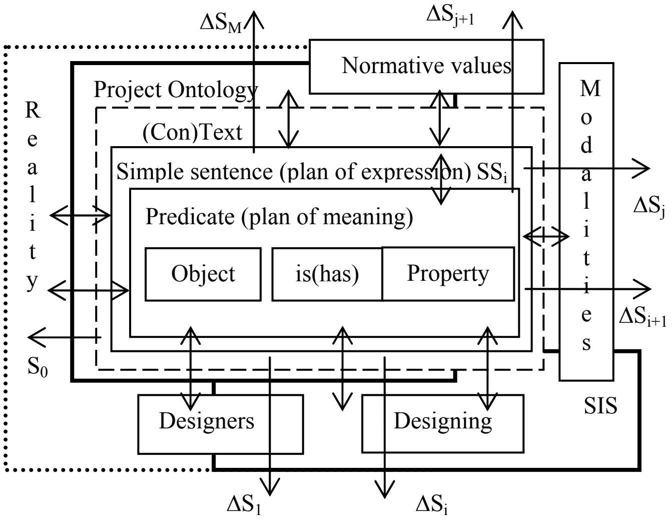


Fig. 4. Model of the simple sentence.

The scheme of relations was used for defining and implementing the techniques for their semantic testing. First of all the expression of semantics for SS was chosen. The structure of the semantics value as a set of semantic components $(S_0 \cup (\cup \Delta S_n))$ is presented generally in Fig. 4 where the component S_0 indicates for the sentence SS its conformity to the reality.

Definition and testing of any other semantic component ΔS_i helps to precise the semantic value of the SS if that can be useful for the design of the SIS. Additionally, the work with any semantic component increases the belief in the correctness of the testable simple sentence (and embedded simple predicate) and can lead to useful questions. In the work with additional semantic components the conditional access to appropriate precedents is used.

Elements of the typical set of semantic components are estimated, applied and tested in the definite sequence. Such work begins from the component S_0 which is compared with elements of the ontology. The result of comparing can be positive or can lead to questions which should be registered.

The positive result does not exclude the subsequent work with additional semantic components.

Semantics of subjectivity and understanding (part ΔS1) are estimated and tested for the relation with designers. The fact of the non-understanding leads to questioning or even to interruption of the work with the testable sentence.

Actual or future material existence of the sentence semantics is a cause for testing the semantic relation of the SS with designing (part ΔSi). Such type of relations is used in the ontology for its systematization.

The greater part of semantic relations of the modality type (parts ΔSi+1 − ΔSj) is aimed to defining and testing of the uncertainties of measurable and/or probable and/or fuzzy types. The semantic relations with normative values (parts ΔSj+1 − ΔSM) suppose the potential inclusion of the SS or its parts into the useful informational sources, for example, into the ontology.

V. Sources of Text Units

As it is shown in Fig. 1 the primary information for filling the project ontology is being extracted by designers from the life cycle of designing the SIS in the real time.

For the designers interaction with the life cycle of the SIS the specialized instrumental system **WIQA** (**W**orking **I**n **Q**uestions and **A**nswers) was created. The main interface of the WIQA is presented in Fig. 5 (with commentary labels).

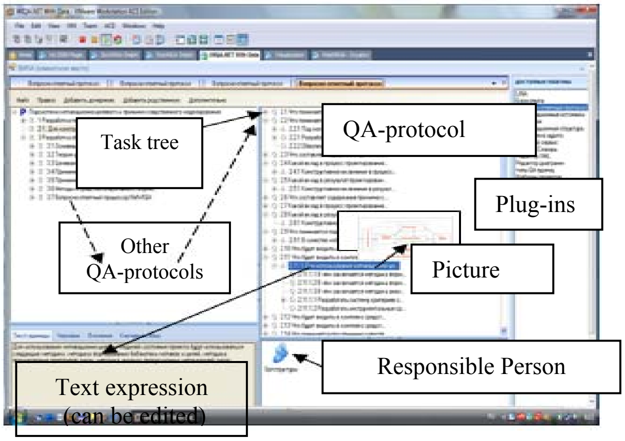


Fig. 5. The main interface of the WIQA.

The WIQA is intended for registering the current state of designing in the form of a dynamic set of project tasks combined into an interactive tasks tree. Each task of such tree is defined with the help of the question-answer protocol of its solving. Any QA-protocol opens the access to the question-answer model (QA-model) of the corresponding task.

The screenshot shows that for the chosen task Z_i from the task tree its QA-model is opened through the QA-protocol of the registered question-answer reasoning (QA-reasoning). Let us notice that any unit of reasoning (question Q_{ij} or answer A_{ij}) has a textual expression with necessary pictures (for example, with UML-diagrams and/or “block and line schemes”). Any task with its statement and any unit of QA-reasoning has the unique name Z.I or Q.J or A.J where I or J is a compound index expressing the subordinations of the corresponding unit. So any text unit is visualized and has a unique index which can be used as its address.

More specifically, any unit of the Z-, Q- or A-type is the interactive object the properties of which are being opened when the special plug-ins are used. One of such plug-ins registers and indicates the responsibility (the assignment of the tasks) in the designer group.

The WIQA is created on the base of the QA-model and the usage of following architectural styles − repository, MVC, client-server and interpreter. So for the current state of design of the definite SIS the WIQA can open to designers the statement of any task from the tasks tree and the definition of any project solution accessible as the definite answer in the corresponding QA-protocol.

Let us notice that the usage of the WIQA as the source of text units is a solution proposed by the author but the suggested ideas are possible to use for creating the project ontology with other instrumental systems which can supply designers by statements of project tasks and definitions of project solutions.

VI. Working Dictionary

The role of the working dictionary is very important in creating the project ontology. This component as the preliminary version of the ontology accumulates all necessary information and distributes informational units between dictionary articles. Carrying out functions of transportation of information, the working dictionary registers relate the text units with their sources. The index name of unit, the number of its sentence and the number of the corresponding simple sentence are used for such referencing.

After extracting the simple sentence with the help of the linguistic processor the predicate model of this sentence is being included into the virtual article of the working dictionary (the article with zero index). Zero article is a temporal memory in the working dictionary which keeps predicates till finishing their testing on the ontology conformity. Zero article, the interface of which is presented in Fig. 6, can be interpreted as a queue of predicates in their mass service.

After extracting any simple sentence and transforming it to the simple predicate, the designer has to start the test of the predicate (as the definite usage of the definite concept). The test begins usually without knowing the “normative usage of the concept” for this predicate in the ontology. Moreover, such usage of the concept in the ontology can be absent or the result of comparing with the appropriate concept will be negative. That is why any tested sentence and corresponding predicate start their life cycles in the working dictionary from zero article.

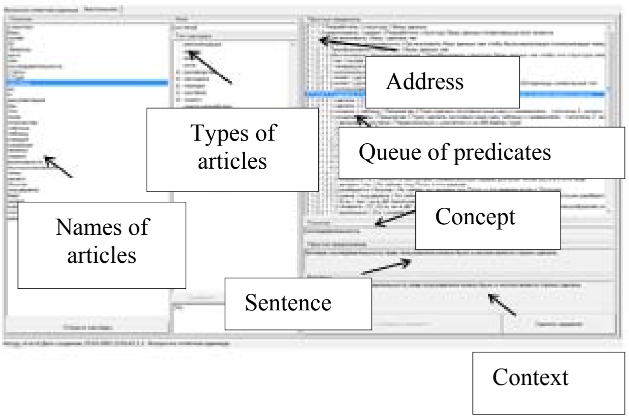
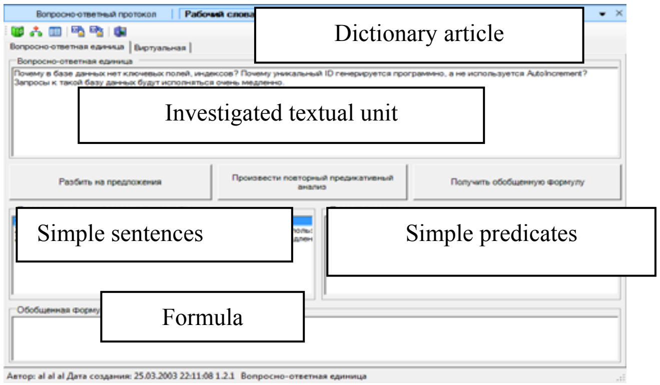


Fig. 6. Virtual article of the working dictionary.

The "normative usage of the concept" for any tested predicate is localized into the corresponding ontology article. If the result of comparing is negative but the designer is convinced that "predicate is truth" then the new ontology article is to be created or the new variant of the concept usage is to be built into the existed ontology article. The first of such results requires to create the new article in the working dictionary also and to transport the tested predicate from the virtual article into this new article (Fig. 7).


Вопросно-ответный протокол
Рабочий слова
Dictionary article
Вопросно-ответная единица
Виртуальная
Investigated textual unit
Разбить на предложения
Произвести повторный предикативный анализ
Получить обобщенную формулу
Simple sentences
Simple predicates
Обобщенная форму
Formula

Вопросно-ответная единица

Почему в базе данных нет ключевых полей, индексов? Почему уникальный ID генерируется программно, а не используется AutoIncrement? Запросы к такой базу данных будут исполняться очень медленно.

Автор: al al al Дата создания: 25.03.2003 22:11:08 1.2.1 Вопросно-ответная единица

Fig. 7. Typical article of the working dictionary.

The type of the new article in the working dictionary is being chosen by designers in accordance with the type of the ontological unit of the designing SIS for representation of which the transported predicate will be used.

Processing the second result includes the transportation of the tested predicate but into the existed article (Fig. 7) of the working dictionary. In general case such predicate is transported into several articles of the working dictionary each of which materializes the tested predicate in the definite form.

If the test of the predicate on the conformity to the ontology is positive then this predicate should be transported in the article of the working dictionary, but only in the article of the definite concept for achieving its representativity. So (step-by-step) predicates (and their parent sentences) are being accumulated into corresponding articles of the working dictionary.

There is a set of types of materialized SIS units which are reflected in the project ontology. The set includes concepts about "parts" of the reality embedded in the SIS and materialized in its software (in the form of variables, classes, functions, procedures, modules, components and program constructions of the other types) and axioms which combine concepts. Each of such unit is found as its initial textual expression in statements of project tasks or in definitions of project solutions. But when this unit is included into the ontology article it is usually rewritten, redefined and reformulated. All informational material for the execution of the similar work is accumulated in the corresponding article of the working dictionary. After creating the adequate textual expressions and formulas they are rewritten from the working dictionary to the corresponding articles of the project ontology.

VII. Logic Processor

The logic processor is intended to build the formal description of the text unit from simple predicates accumulated in the definite article of the working dictionary. Such work is being fulfilled by designer in the operational space presented in Fig. 8 where designer assembles simple predicates in the formula watching them in the graphical window. Necessary predicates are being chosen by designer from the processed article of the working dictionary.

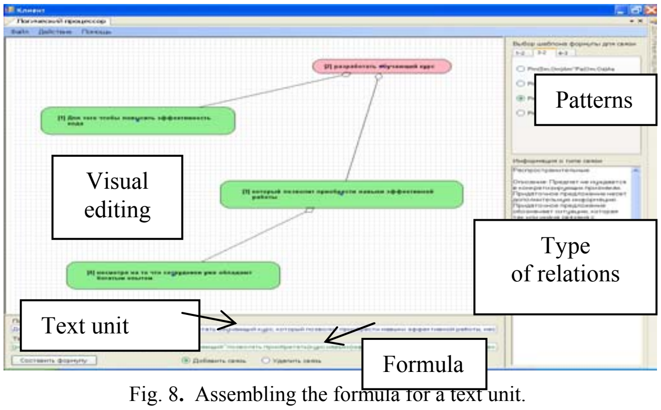


Fig. 8. Assembling the formula for a text unit.

To assemble the predicates the designer has possibilities to use the patterns of two bound predicates and setting of the typical relations between predicates by editing the "picture" (using the drag and drop and lexical information) and registering the final result as the formula of the first predicate logic.

Patterns for two bound predicates has been extracted by author from the grammars of Russian (46 patterns) and English (32 patterns). Such patterns are formalized as typical formulas of the predicates logic.

Mechanisms of assembling the formulas were evolved with experimental aims as the complex of instrumental procedures that provides (for statements of tasks) the creation of prolog-like descriptions. The transformation of the formalized statement of task to the prolog-like description is being implemented as an automated translating of the formula registered in the appropriate article of the working dictionary. Now the method of translating exists in the preliminary version which will be rationalized by the author.

VIII. Systematization of Ontology

The most important feature of any ontology and the project ontology in particular is its systematization. In suggested case the project ontology is defined initially as the Software Intensive System, the integrity of which is provided by the system of architectural views. Some of these views are reflected implicitly by screenshots used in this paper. But such version of the systematization is only one possibility.

Let us present the other way of the systematization. First of all it is the classification of concepts in accordance with structures of the SIS and process of its design. Such system features of the ontology are formed implicitly through definitions of concepts and corresponding axioms.

The next classification level of the ontology is bound with classifying the variants of concept usages. In this case for any concept its article in the project ontology is being formed, which includes the ordered group of concept usage variants and the textual definition of the concept.

The group of usage variants is a list of sub-lists each of which includes main word (or phrase) as a name of the concept (C_i) and subordinated words (or phrases) as names of characteristics (w_{i1}, w_{i2}, …, w_{iN}) of this concept. The definite sub-list w_{i1}, w_{i2}, …, w_{iN}, C_i is an example of the “normative usage of the concept” which can be used in testing of the investigated predicate on the conformity to the ontology.

The basic operation of testing is a comparison of the normative (ontological) sub-list of words with words extracted from the investigated predicate. Two similar sub-lists of words can be extracted from the simple predicate when it indicates the feature and three sub-lists when the predicate registers the relation.

After testing the chosen sub-list of words, which expresses the definite variant of the concept usage, the following results of comparison are possible:

- positive result when the chosen sub-list (w'_{i1}, w'_{i2}, …, w'_{iN}, C_i) is included into the normative sub-list;
- interrogative result when chosen sub-list crosses the normative sub-list or the tested sub-list is outside of all norms (the role of questions was explained above).

The next direction of the systematization is related to binding concepts. For uniting the ontology concepts into the system the following relations are used: basic relations (the part and the whole, the hereditary, the type of the materialization), associative relations (in accordance with the similarity, the sequence, common time and common space) and causality relations.

This type of the view onto the ontology (onto the system of concepts) is formed by administrator of the ontology at the screen shown in Fig. 9. Any unit of any such form is opened for interactive action of designer.

To use the concept relation the designer chooses the necessary concept by its names in the area “keys of entry” and then designer can switch among groups (nodes of the relations system) up to the necessary relation. For the group of relations presented in Fig. 9 the designer may navigate in these directions − “part of”, “whole for”, “has attribute”, “attribute of”, “descendant of”, “parent for”, “has type” and “materialized as”. Similar schemes of navigation are used for the other classes of relations also.

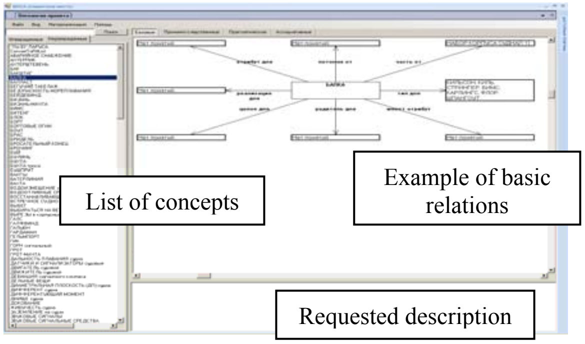


Fig. 9. Systematization of ontology concepts.

In any state of the navigation the description of any visualized unit can be opened. Let us notice that all forms of the ontology systematization are inherited by the working dictionary where it opens the possibility for useful switching between its articles.

IX. Collision Avoidance of Sea Vessels

The proposed version of the project ontology was created and used in the development process of the “Expert system for the collision avoidance of the sea vessels” which is implemented with using the WIQA capabilities [17].

One of the important components of this expert system is a knowledge base which includes the normative rules for the vessel movement. Any unit of such rules was formalized as a precedent with conditional and behavioral parts. Such precedents were extracted from the textual descriptions of normative rules in accordance with their formalizing and coding in expert system by the WIQA capabilities.

At the first stage of the expert system development about 150 textual expressions describing precedents were extracted from 37 rules of The International Regulations for Preventing Collisions at Sea 1972 (COLREGS-72) presented in [3].

Each textual expression was processed with the usage of techniques described above. As a result about 300 concepts with their variants of usages and about 500 precedents were extracted from the textual information. One possibility of the access to the extracted concepts is presented in Fig. 9. Each typical usage of any concept was embedded to the project ontology with its declaration in C#. After developing the expert system the project ontology was refined and included into the created system as its ontology.

As told above all necessary and useful axioms are included into the project ontology also. Any formal expression of any precedent is an axiom binding the definite group of variables indicating the definite concepts.

Each precedent into the project ontology has five variants of these expressions: the textual expression, the predicate formula, the question-answer form, the source code in C# and the executing code. The chosen version of precedent materializations is suitable not only for the automated access by the sailor on duty but for the automatic access of program agents modeling the vessels in the current situation on the sea.

One of these precedents which correspond to the fifteenth rule of MPPSS-72, has the following predicate expression:

if Condition = (Velocity V_1, "keep out of the way")
&& (│Bear_1 - Bear_2│ > 11, 5^o)
&& (CPA-DDA- ΔD1 ≤ 0) then
Reaction = Maneuver_Mi.

The precedent (where CPA is a "Closest Point of Approach", DDA is a normative distance between vessels and ΔD is an error of the distance measuring) is included into the article with as demonstration without full explaining the variables and expressions. The expression of this precedent (as the axiom) is included into the ontology of the expert system for the collision avoidance of the sea vessels.

Let us notice that the set of articles of the project ontology (in development process of the expert system) includes not only units for named variables and precedents. The common quantity of project ontology articles (still under refining) was about two thousand.

X. Conclusion

This paper presents the system of techniques for the creation and usage of the projects ontology in the development of the SIS when enormous quantity of project tasks is being solved by the team of designers in the corporate instrumental network. The success of such activity essentially depends of mutual understanding of designers in their specification of conceptualization for solving project tasks and making project decisions. Therefore any project ontology is to be being created as the dynamic subsystem included into the life cycle of the created SIS.

The main suggestion of the paper is the creation of the project ontology as the problem-oriented $\mathrm{SIS}^{\mathrm{ONT}}$ which is intended for supporting the evolution of understanding and mutual understanding of designers in their step-by-step conceptual activity.

The other important specificity of suggested techniques is the usage of the working dictionary as the preliminary version of the ontology which helps to manage the informational flows and to register the life ways of the informational units and their representativity.

Special attention is given to basic informational units the roles of which are being fulfilled by simple sentences and simple predicates extracted from them. For working with basic informational units the linguistic and logic processors are developed and used. The linguistic processor supports the testing of the statements of project tasks and specifications of project solutions (including requirements and restrictions) on their conformity to the ontology and reality. Arising questions are used for evolving the project ontology.

The logic processor helps to build ontology axioms as predicate formulas. Its experimental research shows that this processor can be (and will be) evolved till the automated creation of the prolog-like description of project tasks.

All interfaces of suggested techniques are adjusted to Russian but only the morphological analyzer and the library of the patterns for two bound predicates are dependent from the specific natural language. The library of patterns for English is created also.

Various and useful techniques of the systematization are embedded into the project ontology for the real time work of designers. Such techniques are accessible both in the ontology component and in the working dictionary.

As the source of the primary information for the creation of the ontology the specialized instrumental system WIQA which supports the usage of question-answer reasoning in the work with project tasks and project solutions is used. Still suggested and developed techniques can be adjusted to the other sources supplying the created ontology by the primary information.

References

[1] H.-J. Bullinger1, J. Warschat, O. Schumacher, A. Slama, and P. Ohlhausen, "Ontology-Based Project Management for Acceleration of Innovation Project," *Lecture Notes in Computer Science,* Vol. 3379, pp. 280-288, 2005.

[2] B. Chandrasekaran, J. R. Josephson1 and V. R.Benjamins, "Ontology of Tasks and Methods," in *Proc. of the Workshop on Applications of Ontologies and Problem-Solving Methods*, held in conjunction with ECAI'98, Brighton, UK, 1998, pp. 31-43.

[3] A.N. Cockcroft, *Guide to the Collision Avoidance Rules: International Regulations for Preventing Collisions at Sea.* Butterworth-Heineman, 2003.

[4] A. H. Eden and R. Turner, "Problems in the Ontology of Computer Programs," *Applied Ontology,* Vol. 2, No. 1, Amsterdam, IOS Press, pp. 13–36, 2007.

[5] A. C. B. Garcia, J. Kunz, M. Ekstrom and A. Kiviniemi, "Building a Project Ontology with Extreme Collaboration and Virtual Design & Construction," *CIFE Technical Report # 152*, Stanford university, 2003.

[6] A. Gelbukh and G. Sidorov, "Approach to construction of automatic morphological analysis systems for inflective languages with little effort," *Lecture Notes in Computer Science*, N 2588, Springer-Verlag, pp. 215–220, 2003.

[7] A. Gelbukh and G. Sidorov, "*Morphological Analysis of Inflective Languages Through Generation*," J. Procesamiento de Lenguaje Natural, No 29, Sociedad Española para el Procesamiento de Lenguaje Natural (SEPLN), Spain, pp. 105-112, September 2002.

[8] A. Gelbukh, G. Sidorov, E. Lavin-Villa and L. Chanona-Hernandez, "Automatic Term Extraction using Log-likelihood based Comparison with General Reference Corpus," *Lecture Notes in Computer Science* 6177, pp. 248-255, 2010.

[9] N. Guarino, "Formal Ontology and Information Systems" in *Proc. of FOIS'98*, Trento, Italy, 1998, Amsterdam, IOS Press, pp. 3-15.

[10] N. Guarino, "Understanding, Building, And Using Ontologies," *Human-Computer Studies*, Volume 46 , Issue 2-3, pp. 293-310, 1997.

[11] N. Guarino, D. Oberle and S. Staab, "What is an Ontology?" in S. Staab and R. Studer (eds.), *Handbook on Ontologies*, Second Edition. International handbooks on information systems. Springer Verlag, pp. 1-17, 2009.

[12] M. Ikeda, K. Seta, O. Kakusho and R. Mizoguchi, "Task ontology: Ontology for building conceptual problem solving models," in *Proc. of*

ECAI98 Workshop on applications of ontologies and problem-solving model, 1998, pp. 126-133.

[13] F. Karray, M. Alemzadeh, J. A. Saleh and M. N. Arab, "Human-Computer Interaction: Overview on State of the Art," *Smart sensing and intelligent systems*, vol. 1, No. 1, pp. 138-159, 2008.

[14] P. Kroll and Ph. Kruchten, *The Rational Unified Process Made Easy: A Practitioners Guide to the RUP*. Addison-Wesley, 2003.

[15] Software Intensive systems in the future. *Final peport ITEA 2 Symposium*, 2006, 68 p. Available: http://symposium.itea2.org/ symposium2006/ main/publications/ TNO_IDATE_study_ ITEA_SIS_in_the_future_ Final_Report.pdf.

[16] P. Sosnin, "Question-Answer Means for Collaborative Development of Software Intensive Systems," *Complex Systems Concurrent Engineering. Part 3*, Springer London, pp. 151-158, 2007.

[17] P. Sosnin, "Question-Answer Expert System for Ship Collision Avoidance," in *Proc. 51th International Symposium ELMAR,* Zadar, 2009, pp. 185-188.

[18] The Standish Group. Available: http://www.standishgroup.com.

Swarm Filtering Procedure and Application to MRI Mammography

Horia Mihail H. Teodorescu and David J. Malan

***Abstract*—Research on swarming has primarily focused on applying swarming behavior with physics-derived or ad-hoc models to tasks requiring collective intelligence in robotics and optimization. In contrast, applications in signal processing are still lacking. The purpose of this paper is to investigate the use of biologically-inspired swarm methods for signal filtering. The signal, in the case of images the grayscale value of the pixels along a line in the image, is modeled by the trajectory of an agent playing the role of the prey for a swarm of hunting agents. The swarm hunting the prey is the system performing the signal processing. The movement of the center of mass of the swarm represents the filtered signal. The position of the center of mass of the swarm during the virtual hunt is reverted into grayscale values and represents the output signal. We show results of applying the swarm-based signal processing method to MRI mammographies.**

***Index Terms*—Swarm intelligence, nonlinear signal filter, MRI, mammography, image processing.**

I. Introduction

SWARMING behavior is widely encountered in populations in nature, where the collective intelligence of the swarm has the advantage of better performing tasks such as escaping from predators, exploring terrain in quest of nutrients, or hunting. Swarming is the collective, aggregated, corroborated behavior of a set of individuals (agents) of similar or identical structure, each of them able to sense, move, and make decisions of their own while communicating with the other agents or while observing the other agents' movements. A swarm is said to possess a collective (distributed) intelligence because a specific behavior occurs at the group level. While each agent potentially has full autonomy and decision-making capability, their individual behaviors are aggregated through the coupling of the movements of neighbor agents. This new behavior results from the aggregation of individual movements. Consequently, the displacement of an individual is largely decided at the group level, allowing a swarm to accomplish tasks that would be unreachable to a single agent, or to better accomplish such tasks. Artificial swarms mimic the behavior of groups of insects, fish schools, or herds of animals.

The purpose of this paper is to introduce and demonstrate a novel method of nonlinear filtering based on swarm dynamic. We apply the method to MRI images with the goal of testing the capabilities of swarm processing. The method we propose transforms an image into a surrounding relief that directly modifies the swarming behavior during hunting. The swarm views the image as a 'ground surface' it flies over. Each agent in the swarm sees the closest pixels as constraints of the movement. We model the signal (process) to be filtered by the dynamic (trajectory) of an agent playing the role of prey. A swarm comprising several hunting individuals hunts a prey, according to a modified swarm procedure. Thus, when the prey follows a line in the image, the swarm averaged trajectory processes a line of the image, filtering the image line and re-morphing it. The resulted filters will be generically named swarm filters. The number of individuals in the swarm and various parameters of the swarm constitute the parameters of the filter. The output (filtered) signals are obtained as the average accelerations of the swarm in the x, y, and z directions of motion. The model we adopted is motivated by the need to create a direct interaction between the swarm and the image, still preserving the main concepts in the swarming theory and a reasonable resemblance with the natural swarming processes. Our hypothesis, which is verified by the preliminary results, was that the swarm 'smoothes', that is, filters and morphs the relief underneath it. At this stage, we manually optimized the type of interactions between the agents and the image 'relief'.

The swarm signal processing pertains to the larger class of nonlinear methods. The nonlinearity of the swarming is produced by the nonlinear terms in the swarm movement equations we use, in the first place the distance-dependent terms, as presented in Section 2. The swarm processing has a different mechanism of operation than other nonlinear signal processing systems, such as neural networks, statistical filters, and dynamic range compression filters. The swarm processing can be thought of as a nonlinear procedure based on a set of nonlinear, second order coupled equations, each describing the movement of an element of the swarm under the constraints imposed by the signal impersonated by the prey.

The swarm model we propose for signal processing combines features from several swarm models presented in the literature and uses a few new constraints and characteristics. Subsequently, we briefly recall several models on which ours relies. Physics-derived models, as described by Elkaim *et al.* [1], often focus on determining discrete models for the positions and speeds of the particles, given close-contact forces between swarm agents, such as spring-type forces in the models of Elkaim [1], [2], and a stronger force to a leader. Olshevsky *et al.* [3], and Olfati-Saber and Murray [4] describe

Manuscript received June 8, 2010. Manuscript accepted for publication July 25, 2010.

The authors are with Harvard University, USA (hmteodor@ieee.org).

another set of models involving node-to-node interactions on a graph network. A consensus in the literature is that a constraint in swarm-type applications is communication bandwidth; as a consequence, swarm agents have some form of very elementary memory. Olfati-Saber and Murray [4], [5] propose Markov type I and II models. In many previous models, the accelerations of the swarm agents can have large discontinuities. Some swarm models, illustrated by Elkaim's models in particular allows the swarm the possibility of easily splitting into subgroups after avoiding an obstacle. The use of the center of mass as a virtual leader, as well as the use of the same type of spring force for interactions member-member and member-obstacle contributes to this behavior. We included in our proposed discrete models limits for the allowed accelerations. Also, we use an equation for acceleration that favors more the coherence of the swarm.

Swarming algorithms are strongly dependent on the neighborhood concept and involve the distance between agents. While there is substantial evidence in nature that communication between swarm members is not only through visual interaction, the implicit assumption in the modeling literature is that the distance perception is Euclidean (as in the works of Elkaim *et al.* [2], Murray *et al.* [4]). Anstey *et al.* have shown in a recent Science article [6] that the communication between swarm members that triggers gregarization (or swarming behavior) is done through chemical recognition (smell). Use of communication through odor is also found in populations of wasps [7], of spiders [8], and of bees [9]. Other forms of non-visual communication appear in populations of fish (swarming communication through odor, according to Todd *et al.* [10]) and frogs (evidence of ultrasonic communication is shown in the works of Feng *et al.* [11], and of Arch *et al.* [12]). Therefore, there is no imperative requirement for the usage of Euclidean distance for modeling perception for every application; consequently, we used models based on both Euclidean and non-Euclidean metrics.

Ant colony algorithms are a fundamentally different approach than that of physics-derived swarming models discussed above. Ant colony swarming occurs based on evolution of pheromone fields based on trails left by individual agents, as opposed to neighboring agent interactions. Recently, Ramos *et al.* [13], [14], Huang *et al.* [15], [16], and Ma *et al.* [17] applied ant colony algorithms to image processing for medical images. These authors used the pheromone traces left by the ants to create contour-like images superposed over the initial images. Contour enhancement is not optimal for MRI images due to the gradient variations between areas in the image as opposed to sharper differences between neighboring image areas and an alternate method to ant colony algorithms has to be developed. In our approach, we prefer the swarming, without feromone-like memory, as the model that implements the signal processing.

The organization of the paper is as follows. In the second section we present the details of the proposed swarming model and of its implementation. We present a few image processing results in the third section. The last section comprises conclusions.

II. Proposed Swarming Model

We propose a three-dimensional swarm dynamic based on neighbor interactions dependent on nonlinear attraction and repulsion forces. The forces are piece-wise functions dependent on the vicinity criteria. The purpose of the attraction and repulsion forces is to mediate swarm aggregation. Literature models such as those of Elkaim *et al.* [1], [2] propose elastic forces both for attraction and repulsion.

A. General Setting

We say that the vicinity of agent i is dynamic if the neighbors j of i can change over time. The set of neighbors V_i of agent i is recomputed for every timestep. We determine the dynamic vicinity using a fixed-radius model in which the neighbors j of i are determined based on the condition that the distance between the two agents satisfies $d(i,j) \leq \rho$ where we consider ρ to be the radius of a spherical vicinity having agent i at its center. Neighbors who are closer to i than the distance d_{min} are subject to a repulsion force; otherwise neighbors are subject to an attraction force.

While attraction and repulsion forces are responsible for the internal cohesion of the swarm, the swarm is not driven across the input image by the internal forces. An external force is necessary to drive the swarm across the input image. Unlike literature models such as those of Murray *et al.* [4] and Elkaim *et al.* [1] where the purpose of the swarm dynamic is given by a constant bias in the velocity of the swarm agents, our model is a predator-prey model in which the swarm of predator agents is attracted to a prey by an agent-prey force that acts independently on every agent in the swarm. This is a forcing factor in the equations of motion of the swarm. We also consider a friction force that depends on the speed of the agent. Therefore, in the model we propose there are three types of forces that act upon a given swarm agent i: internal swarm attraction/repulsion forces between i and all its neighbors j, external force acting upon agent i from the prey p, and friction forces. The force, $F_x(i)$, acting on agent i in the x direction of motion is the result of:

$$\begin{aligned}\vec{F}_x(i) = \vec{F}_{friction,x}(i) + \\ + \sum_{j \in V_i} \vec{F}_{internal,x}(i,j) + \vec{F}_{external,x}(i,p)\end{aligned} \tag{1}$$

where the notation $j \in V_i$ signifies that we take the contributions from all neighbors j in the vicinity V_i of agent i and the notation (i,j) signifies that the force depends on agents i and j. The general equation above holds also for the y and z directions of motion.

The internal cohesion of the swarm is managed by agent-to-agent attraction and repulsion forces which act upon neighboring agents. In order to prevent collisions among neighboring agents, a repulsion force acts upon two neighboring agents if the distance between them is less than a threshold $d_{min} < \rho$. An attraction force acts upon neighboring agents otherwise. We chose the following internal cohesion force, where the coefficient of the repulsion force is β and the coefficient of the attraction force is α:

$$\vec{F}_{internal}(i,j) = \begin{cases} \frac{\beta \cdot (\vec{r_i} - \vec{r_j})}{(r_i - r_j)^3} & \text{if } |\vec{r_i} - \vec{r_j}| \le d_{min} \\ \frac{\alpha \cdot (\vec{r_i} - \vec{r_j})}{(r_i - r_j)^4} & \text{if } d_{min} < |\vec{r_i} - \vec{r_j}| \le \rho \end{cases} \tag{2}$$

where $\vec{r_i}$ denotes the position vector of the agent i. The overall swarm cohesion force acting on agent i due to its neighborhood is $\sum_{j \in V_i} \vec{F}_{internal}(i,j)$. Note that the distance function $d(i,j) = |(\vec{r_i} - \vec{r_j})|$ may be chosen as non-Euclidean. In this section we only treat the Euclidean distance case. In subsection 2.5 we briefly discuss a non-Euclidean distance example. The force acting on agent i in the x, y, and respectively z directions due to the neighborhood of i are:

$$\begin{aligned} \vec{F}_{neighborhood,x}(i,t) &= \sum_{j \in V_i} \vec{F}_{internal}(i,j) \cdot \\ &\cdot \frac{x_i(t-1) - x_j(t-1)}{d(i,j)} \\ \vec{F}_{neighborhood,y}(i,t) &= \sum_{j \in V_i} \vec{F}_{internal}(i,j) \cdot \\ &\cdot \frac{y_i(t-1) - y_j(t-1)}{d(i,j)} \\ \vec{F}_{neighborhood,z}(i,t) &= \sum_{j \in V_i} \vec{F}_{internal}(i,j) \cdot \\ &\cdot \frac{z_i(t-1) - z_j(t-1)}{d(i,j)} \end{aligned} \tag{3}$$

where by the notation (i,t) we refer to the agent i at current moment of time t.

We choose the following velocity-dependent friction force that acts on agent i at time t:

$$\begin{aligned} \vec{F}_{friction,x}(i,t) &= -\mu \cdot \dot{x}_i(t-1) \\ \vec{F}_{friction,y}(i,t) &= -\mu \cdot \dot{y}_i(t-1) \\ \vec{F}_{friction,z}(i,t) &= -\mu \cdot \dot{z}_i(t-1) \end{aligned} \tag{4}$$

where we consider the friction coefficient μ to be a constant and the same for all agents in the swarm and is in the range $0 \le \mu \le 1$. A higher value of μ implies a faster damping of the movement. Physically the choice of the friction force is appropriate for the motion of the agent in an idealized fluid of low viscosity and with no turbulence.

B. *Interaction with the Image*

The external force acting on every agent of the swarm is an elastic force that sets the goal for the swarm to follow the trajectory of the prey p. From a physical point of view, this agent-prey force ensures that every agent is attracted to the prey. We choose a prey-agent force of the following expression:

$$\begin{aligned} \vec{F}_{external,x}(i,p,t) &= \lambda_x \cdot (x_p(t-1) - x_i(t-1)) \\ \vec{F}_{external,y}(i,p,t) &= \lambda_y \cdot (y_p(t-1) - y_i(t-1)) \\ \vec{F}_{external,z}(i,p,t) &= \lambda_z \cdot (z_p(t-1) - z_i(t-1)) \end{aligned} \tag{5}$$

where $\lambda_x, \lambda_y, \lambda_z$ are coefficients. The use of the prey-agent force in the image processing algorithm is discussed in sect. 2.6.

C. *Discretizing the Movement Equations*

We will discretize the equations of motion of the swarm agents with a timestep parameter, δ, which we vary in the simulations section. The equations of motion in the x, y, and respectively z direction for agent i are:

$$\{x_i(t) = x_i(t-1) + \delta \cdot \dot{x}_i(t),\ y_i(t) = y_i(t-1) + \delta \cdot \dot{y}_i(t), \\ z_i(t) = z_i(t-1) + \delta \cdot \dot{z}_i(t)\}$$

where the equations of the speeds in the directions of motion are:

$$\{\dot{x}_i(t) = \gamma \cdot \dot{x}_i(t-1) + \delta \cdot \ddot{x}_i(t),\ \dot{y}_i(t) = \gamma \cdot \dot{y}_i(t-1) + \delta \cdot \ddot{y}_i(t), \\ \dot{z}_i(t) = \gamma \dot{z}_i(t-1) + \delta \cdot \ddot{z}_i(t)\}$$

where the term in the coefficient γ was introduced by Olfati-Saber and Murray [5] and represents a form of elementary memory of the agents (Markov I type relationship in the speed variation equations). The acceleration of the agent i depends on the neighborhood force (3) acting on i, the agent-to-prey force (5) on i, and the friction force (4) of i:

$$\begin{aligned} \ddot{x}_i(t) = (\gamma - 1) \cdot \dot{x}_i(t-1) + \\ + \sum_{j \in V_i} \vec{F}_{internal}(i,j) \cdot \frac{x_i(t-1) - x_j(t-1)}{d(i,j)} + \\ + \lambda_x \cdot (x_p(t-1) - x_i(t-1)) - \mu \cdot \dot{x}_i(t-1) \\ \ddot{y}_i(t) = (\gamma - 1) \cdot \dot{y}_i(t-1) + \\ + \sum_{j \in V_i} \vec{F}_{internal}(i,j) \cdot \frac{y_i(t-1) - y_j(t-1)}{d(i,j)} + \\ + \lambda_y \cdot (y_p(t-1) - y_i(t-1)) - \mu \cdot \dot{y}_i(t-1) \\ \ddot{z}_i(t) = (\gamma - 1) \cdot \dot{z}_i(t-1) + \\ + \sum_{j \in V_i} \vec{F}_{internal}(i,j) \cdot \frac{z_i(t-1) - z_j(t-1)}{d(i,j)} + \\ + \lambda_z \cdot (z_p(t-1) - z_i(t-1)) - \mu \cdot \dot{z}_i(t-1) \end{aligned}$$

D. Remarks on Movement Equations

We recognize that the equations of motion of the agent i are forced second-order nonlinear differential equations where the forcing term is given by the trajectory of the prey. Since the equations depend on the agent index i, the dynamic of the swarm is that of a coupled system of nonlinear differential equations of the following form:

$$\begin{aligned} \{\ddot{x}_i + \varphi(\dot{x_i}) + \Psi(x_i, x_j) &= \lambda_x \cdot x_{prey}(t) \\ \ddot{y}_i + \varphi(\dot{y_i}) + \Psi(y_i, y_j) &= \lambda_y \cdot y_{prey}(t) \\ \ddot{z}_i + \varphi(\dot{z_i}) + \Psi(z_i, z_j) &= \lambda_z \cdot z_{prey}(t)\}_{1 \le i,j \le N} \end{aligned} \tag{6}$$

where the coupling is done by the term $\Psi(x_i, x_j)$ (the swarm cohesion forces) and N is the number of agents in the swarm. We take as initial conditions the positions, speeds, and accelerations of all swarm agents at timestep $t = 0$. The average of the accelerations $\ddot{x}_i$, $\ddot{y}_i$, $\ddot{z}_i$ for every timestep t represent the pixels of the three output images. The image-swarm interaction is discussed in detail in the following section. Notice that linearizing the coupled system of differential equations (6) we would obtain a system of coupled linear, damped, oscillators. The local behavior would be similar to that of the model by Elkaim *et al.* [1] in that locally the forces would be elastic.

E. Non-Euclidean Swarm Distance Perception

We introduce a swarm dynamic with non-Euclidean distance function instead of the Euclidean $d(i, j)$ used in the previous sections. While the use of Euclidean distance in literature seems natural for determining agent interactions, this choice is not justified by biological facts. In biological swarming, the distances between individuals are determined through a means of communication that allows swarm members to determine their relative distance, based on a senses, such as odor, hearing, and vision. Recent research on locust swarming showed that odor was mediating neighbor interactions [6]. There is no evidence that odor is producing measurements based on an Euclidean distance. We departed from the biological swarming models, that typically use Euclidean distance by using non-Euclidean distance metrics might be valuable for image enhancement using swarm processing algorithms. As an example, we used a logarithmic distance function and applied the resulting swarm filter to the two input images. The logarithmic distance function we used is:

$$d(i,j) = \log\left(1 + \eta \cdot \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2) + (z_i - z_j)^2}\right)$$

where we used values of η ranging from 1 to 10 in increments of 1 per simulation. The output images using the logarithmic distance did not show any improvement over the input images. Nonetheless, we consider the analysis of non-Euclidean distances for the swarm algorithm to be a direction for future work, specifically designing distance functions that enable the emphasis of desired features in the input images.

F. Image Processing Algorithm and Implementation Details

An image is represented by a matrix M indexed over $i, j \in \mathbb{N}$, where (i, j) represent the positions of the pixel (range $0 \le i \le x_{max}$, $0 \le j \le y_{max}$), and the values in the matrix represent grayscale levels (range 0 to 255). The number of timesteps used in the simulation corresponds to the number of pixels $x_{max} \cdot y_{max}$ in the input image. Each timestep t corresponds to a different pixel in the image, as follows: starting with the first horizontal line in the input image($i = 0$), traverse each line in the input image in order left-to-right $j = 0$ to $j = y_{max} - 1$ and store the graycale value of the current pixel (i, j) corresponding to timestep $t = i \cdot y_{max} + j$ into the prey's x, y, and respectively z coordinate values:

$$\{x_p(t) = \chi_x \cdot M(i,j),\ y_p(t) = \chi_y \cdot M(i,j), \\ z_p(t) = \chi_z \cdot M(i,j)\} \tag{7}$$

where $M(i, j)$ denotes the grayscale value for pixel (i, j). The use of different coefficients χ_x, χ_y, χ_z coefficients in (7) allows the application of three different swarm filters on the same input image. Computing the average of the swarm agents' accelerations at every timestep t for the x, y, and z directions of motion allows the output of three images that are obtained from converting the $\ddot{x}(t)_{avg}$, $\ddot{y}(t)_{avg}$, $\ddot{z}(t)_{avg}$ values of the swarm into grayscale values. If the values are above the grayscale value of 255 they are truncated to 255 in the resulting image (the same holds for negative values which we truncate to a grayscale value of 0). The resulting images varied significantly based on the values of the χ coefficients.

The swarm's positions, speeds, and accelerations are pseudorandomly generated for $t = 0$ as initial conditions. The values of all parameters in the algorithm are fixed at the initialization step. Also in the initialization step all values of x_p, y_p, and z_p are computed and stored. As described in sect. 2.1, the accelerations of a given agent i depend on determining the forces acting upon i from its neighbors $j \in V_i$. The implementation has been done in *C*.

III. Results

For determining the power and the limits of the swarm image processing, we performed simulations on two abnormal Mammographies from the NIH [18] (see Fig. 1). The area of interest in the input images consists of the calcified deposits. The calciferous deposits produced by cancer are depicted in the input images by an arrow. The problem in mammography imaging is to de-blur the image and to eliminate the useless details for evidencing the micro-calcifications. Typical image enhancement procedures, like contrast manipulation and histogram equalization are only partly effective in this respect, as they emphasize various useless details at the same time with the micro-calcifications (see Fig. 4). In fact, histogram equalizers may further mask the calcification into the details of the scene, as in Fig. 5. In contrast, after the swarm filtering, the elements of interest are shown over a flat, almost uniform gray surrounding. The nonlinear filter emphasizes these Ca deposits as in Fig. 3. However, the swarm processing results depend on the processed image and may produce results similar to the ones obtained based on histogram manipulation. The case is exemplified in Fig. 1 and Fig. 5.

During the simulations, we adjusted the parameters of the algorithm in order to emphasize the deposits with respect to the surrounding tissue. We produced output images either using the acceleration of the center of mass of the swarm or the speed of the center of mass of the swarm as the filter (see Fig. 3, Fig. 1).

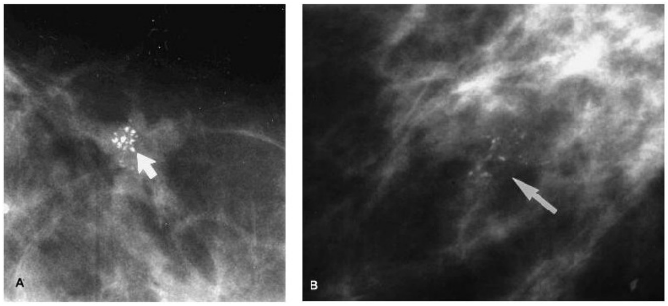


Fig. 1. Input Mammographic Images A and B, adapted from NIH [18], used to demonstrate the nonlinear filter.

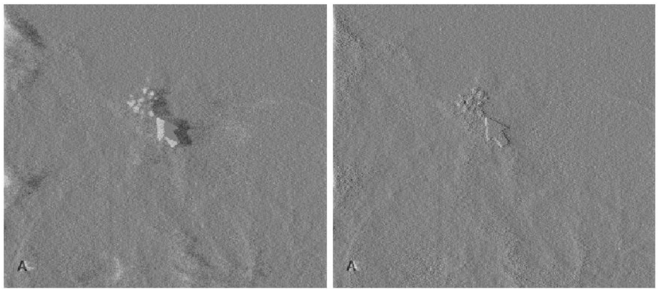


Fig. 2. Feature extraction for input image A using the z-acceleration (left panel) and x-acceleration (right panel) of the center of mass of the swarm as differentiator filters and $\chi_x = 30$, $\chi_z = 10$.

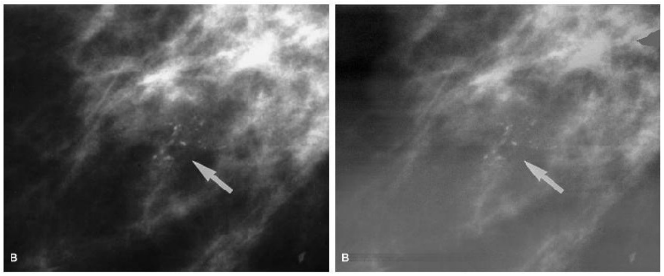


Fig. 3. Original input image B (left panel) versus filtered image using the z-acceleration of the center of mass (right panel) using $\chi_z = 1$.

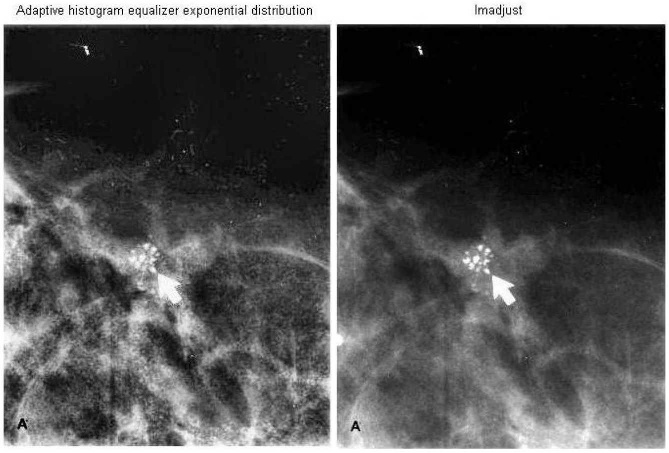


Fig. 4. Results obtained with histogram manipulation on the first mammographic image (left panel) and contrast adjustment (right panel)

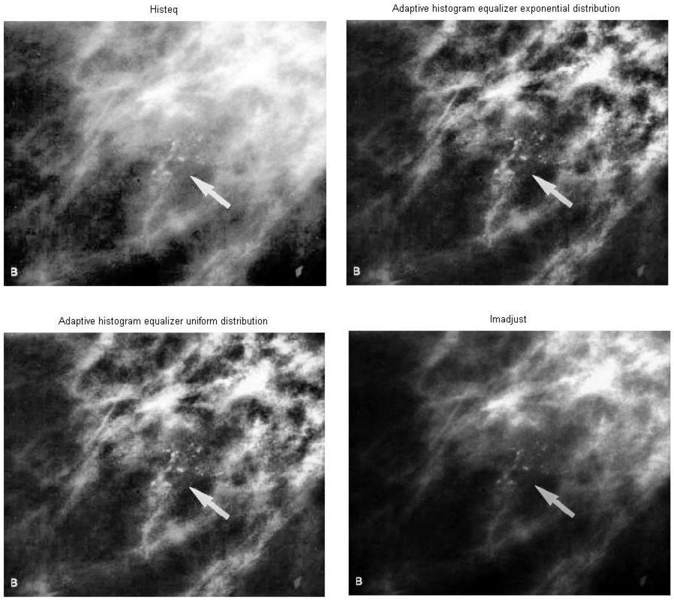

Histeq
Adaptive histogram equalizer exponential distribution
B
B
Adaptive histogram equalizer uniform distribution
Imadjust
B
B

Fig. 5. Results obtained with histogram manipulation on the second mammographic image

We performed numerous simulations on the input images to empirically determine ranges of values for the parameters that resulted in usable output images. We varied the swarm size between 5 agents, which we experimentally determined to be the minimum size of swarm that would affect the input image, and 50 agents in increments of 5. Maintaining all other parameters constant, swarms larger than 25 agents had the same effect on the input image as swarms of size 25 agents. The parameter γ had a significant impact on the output image. A value of γ larger than 1 resulted in a uniform gray image (no features), while values of γ under 0.5 in decrements of 0.1 decreased the contrast of the image until no features were visible for γ of 0.1. We obtained the best results regarding emphasizing the features of interest in the image with values of γ between 0.9 and 0.98. The friction coefficient μ also had a significant impact on the result image. We varied μ in increments of 0.1 from 0 to 1.5. No features were obtained for μ of 0 and for μ above 1. The neighborhood threshold distance ρ variation from 1 to 150 had no significant impact on image quality. The value of δ used in the output images Fig. 2 was of 0.15 and in Fig. 3 was of 0.0005 and the number of agents in the swarm used for both figures was 25. The coefficients λ_x, λ_y, λ_z, α, and β were set at the value of 1 throughout the simulations. A major impact on the output images was due to the χ_x, χ_y, χ_z coefficients, which, when distinct from each other, would generate three distinct output images. An example of this effect is shown in Fig. 2, where the two distinct images were obtained for the same set of parameters and same input image, except for $\chi_x \neq \chi_z$ and resulted in different filtering applied in the swarm's x and respectively z directions of motion. Since the swarm dynamic is highly nonlinear, changing one of the three χ coefficients while keeping the other two constant as in a previous run will change all three output images. The values used to illustrate this difference between the x output image and the z output image in Fig. 2 were $\chi_x = 30$, $\chi_y = 5$, and $\chi_z = 10$. The values of $\chi_x = 30$, $\chi_y = 5$, $\chi_z = 1$ are used in Fig. 3.

IV. Conclusions

While image processing, including feature emphasis, has recently benefited of many artificial intelligence techniques such as neural networks and genetic algorithms, not to mention various combinations of statistic filters, tasks like mammographic image processing remained unsatisfactory solved. In the research reported here, we devoted ourselves to conceiving a method that could make use of the elementary collective intelligence of the swarms to tackle image processing. We introduced a novel nonlinear swarm dynamic that incorporates non-Euclidean agent distance perception and which we apply to image processing. While we exemplified feature emphasis on MRI mammography images, the method could be extended for feature emphasis in other classes of images and be trained in view of feature recognition.

The interaction forces between the agents and the image relief that we used in this paper may be somewhat unnatural for a biological swarm, hence the research should continue in identifying new types of interaction forces that, on one side, are closer to the natural ones and, on the other side, preserve or enhance the filtering results.

The research presented is incipient. Extensive tests must be performed on a variety of images under a large spectrum of noises to check the robustness of the filters proposed. Also, the quality of the results was not the same for different classes of processed images, each class requiring a trimming of the swarm coefficients to perform well or at least acceptably. Thus, we need to derive an automated optimization procedure of the swarm processing system before the method can see extended use. Future collaboration with experts in the medical field is essential in determining the degree of utility of the swarm signal processing in mammography and in other imaging fields.

Acknowledgements

The first author is indebted to the organizers of the MMSB Workshop in Israel, January 2010 for a grant for participation in the workshop and especially to Professor Urszula Ledzewicz for encouragements to continue the work.

References

[1] G. H. Elkaim and M. Siegel, "A lightweight control methodology for formation control of vehicle swarms," in *Proc. IFAC 2005*, Prague, July 4-8 2005.

[2] G. H. Elkaim and R. J. Kelbley, "Extension of a lightweight formation control methodology to groups of autonomous vehicles, in *Proc. ISAIRAS 2005*, Munich, September 5-9 2005.

[3] A. Olshevsky and J. N. Tsitsiklis, "Convergence rates in distributed consensus and averaging," in *Proc. IEEE 45th Conference on Decision and Control*, pp. 3387-3392, December 13-15 2006.

[4] R. Olfati-Saber and R.M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. on Automatic Control*, Vol. 49, No. 9, September 2004.

[5] R. Olfati-Saber, J.A. Fax, and R.M. Murray, "Consensus and cooperation in networked multi-agent systems," in *IEEE Proceedings*, Vol. 95, No. 1, pp. 215-233, January 2007.

[6] M. L. Anstey, S.M. Rogers, O. M. Swidbert, M. Burrows, and S.J. Simpson, "Serotonin mediates behavioral gregarization underlying swarm formation in desert locusts," *AAAS Science*, Vol. 323, pages 627–630, January 30 2009.

[7] M. G. Naumann, "Swarming behavior: evidence for communication in social wasps," *AAAS Science*, Vol. 189, pp. 642–644, August 22 1975.

[8] Y.D. Lubin, "Dispersal by swarming in a social spider," *AAAS Science*, Vol. 216, pp. 319–321, April 16 1982.

[9] R. A. Morse "Swarm orientation in honeybees," *AAAS Science*, Vol. 141, pp. 357–358, July 26 1963.

[10] J. H. Todd, J. Atema, and J. E. Bardach, "Chemical communication in social behavior of a fish, the Yellow Bullhead (Ictalurus natalis)," *AAAS Science*, Vol. 158, pp. 672–673, November 3 1967.

[11] A.S. Feng, P.M. Narins, C.-H. Xu, W.-Y. Lin, Q. Qiu, and Z.-M. Xu, "Ultrasonic communication in frogs," *Nature Letters*, Vol. 440, pp. 333–336, March 16 2006.

[12] V.S. Arch, T. U. Grafe, and P.M. Narins, "Ultrasonic signaling by a Bornean frog," *Biology Letters*, Vol. 4, pp. 19–22, February 23 2008.

[13] V. Ramos, F. Muge, and P. Pina, "Self-organized data and image retrieval as a consequence of inter-dynamic synergistic relationships in artificial ant colonies," in *Javier Ruiz-del-Solar, Ajith Abraham and Mario Kppen (Eds.), Frontiers in Artificial Intelligence and Applications, Soft Computing Systems - Design, Management and Applications, 2nd Int. Conf. on Hybrid Intelligent Systems*, IOS Press, Vol. 87, pp. 500–509, Santiago, Chile, December 2002.

[14] V. Ramos and F. Almeida, "Artificial ant colonies in digital image habitats - a mass behavior effect study on pattern recognition," arxiv.org/abs/cs/0412086.

[15] P. Huang, H. Cao, and S. Luo, "An artificial ant colonies approach to medical image segmentation," *Computer Methods and Programs in Biomedicine*, Vol. 92, pages 267–273, Elsevier, 2008.

[16] T. Huang and A.S. Mohan, "A Microparticle Swarm Optimizer for the Reconstruction of Microwave Images," *IEEE Transactions on Antennas and Propagation*, Vol. 55, Issue 3, pp. 568–576, March 2007.

[17] L. Ma, K. Wang, and D. Zhang, "A universal texture segmentation and representation scheme based on ant colony optimization for iris image processing," *Computers and Mathematics with Applications*, Vol. 57, pages 1862–1868, Elsevier, 2009.

[18] J. Holland and E. F. Frei, "Cancer Medicine," *NIH NCBI Bookshelf*, Fifth Edition, Chapter 30, Fig. 30F.12, March 2007.

Mixing Theory of Retroviruses and Genetic Algorithm to Build a New Nature-Inspired Meta-Heuristic for Real-Parameter Function Optimization Problems

Renato Simões Moreira, Otávio Noura Teixeira, and Roberto Célio Limão de Oliveira

***Abstract*—This paper describes the development of a new hybrid meta-heuristic of optimization based on a viral lifecycle, specifically the retroviruses (the nature's swiftest evolvers), called Retroviral Iterative Genetic Algorithm (RIGA). This algorithm uses Genetics Algorithms (GA) structures with features of retroviral replication, providing a great genetic diversity, confirmed by better results achieved by RIGA comparing with GA applied to some Real-Valued Benchmarking Functions.**

***Index terms*—Evolutionary computation, genetic algorithm, viruses, retroviruses, hybrid metaheuristic.**

I. Introduction

OVER the years the viruses have been treated as villains in the destruction of organic structures, resulting from the disappearance of entire species and even causing long and fatal epidemics (such as HIV). However, its effectiveness to perpetuate themselves is really impressive, although their acceptance as life forms are still under debate. Retroviruses are the nature's swiftest forms [1] and this retroviral feature could not be discarded to develop some computational structure (specifically in the field of evolutionary computation) that uses them as inspiration.

This manuscript work will describe the development of a new metaheuristic structure inspired on viral structures of family Retroviridae (retroviruses), this algorithm is called as Retroviral Iterative Genetic Algorithm (RIGA). The source of the name comes from the junction of its features: *Genetic Algorithm* for behaving like a GA, *Retroviral* for having retroviruses structures and *Iterative* because occurs every single generation.

II. Biological Basement: from Viruses to Retroviruses

Viruses are compulsory intracellular parasites with a very simple structure. Their acceptance as life forms is very controversial, since they are very different from the most simple bacteria and they have unique features, like the absence cell membrane, they don't have any known organelles and their size is several smaller, thus, the only possible way to see them is by electronic microscopy. They are also metabolically inert unless they are inside a host cell. It is important to notice also that they cannot contain simultaneously DNA and RNA molecules [3].

The viruses are formed basically by two components: the capsid, consisting of viral proteins, and the core, which contains their genetic information; the combination of these two structures is known as nucleocapsid. The main objective of viruses is to replicate themselves. To achieve this, they need to penetrate a host cell, make copies of themselves and put those copies out of the host cell.

A. Retroviruses

Retroviruses are the only known entities that are able to convert RNA into DNA under normal circumstances. After the adsorption and the injection of their genetic material into the host cell, the process of retrotranscription takes place in the cytoplasm of the infected cell, using the viral reverse transcriptase enzyme. This process will convert a single-stranded molecule of RNA (ssRNA) into a double-stranded DNA (dsDNA) molecule that is larger than the original RNA and has a high error rate, creating DNAs sequences different of which should be [4].

The retroviruses replication process can be described basically in follow steps [1]:

– Viral recognition by the receptors present in the host cell surface;
– Penetration into the host cell;
– Reverse transcription (RNA to DNA)
– Viral integration to the host's genome, where it will replicate;

Manuscript received May 7, 2010. Manuscript accepted for publication August 29, 2010.

Renato Simões Moreira and Roberto Célio Limão de Oliveira are with PPGEE-ITEC, Universidade Federal do Pará (UFPA), Belém, PA, Brasil (renatosm@gmail.com; limao@ufpa.br).

Otávio Noura Teixeira is with Laboratório de Computação Natural (LCN), Centro Universitário do Pará (CESUPA), Belém, PA, Brasil; Movimento Evolucionário e Cooperativo para a Construção do Artificial (MEC2A), Belém, PA, Brasil; PPGEE-ITEC, Universidade Federal do Pará (UFPA), Belém, PA, Brasil (onoura@gmail.com).

- Viral DNA translation (produces viral mRNA that will translated in viral proteins)
- Viral assembling
- Viral shedding, when the new viruses leave the host cell.

One of the proteins of the virus is the integrase, which is still associated with provirus. This enzyme cuts the chromosomal DNA of the host cell and inserts the viral-converted DNA, integrating the provirus into the host cell chromosome (Fig. 1.). The next time this infected cell divides, the provirus will be replicated to the daughter cells [1]. After the viral genome is integrated in the host cell genome, the virus will be totally dependent of the cellular metabolism to continue its process of transcription, translation, genome replication, viral assembling and shedding.

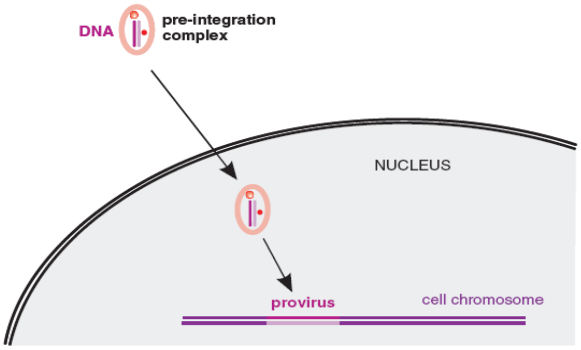


Fig.1. Provirus creation and reverse transcriptase.

The concepts of Charles Darwin about reproduction and Natural Selection applied to organics forms are also applied to viruses. Even though their acceptance as an organic form is very questionable, the viruses have genes that are striving to perpetuate the species. The main mechanisms used for viral evolution are mutation, recombination, reassortment and acquisition of cellular genes [1].

III. Genetic Algorithms

Genetic algorithms are part of probabilistic techniques and may find different solutions in different executions with the same parameters, even with the same population [5]. Some of the main advantages of GA are [2]:

- Optimization of discrete and continues values.
- Simultaneous search.
- Possibility to work with many variables.
- Provide a set of solutions, instead of only one specific solution.

The fundamental principle is to explore a space of search by a population of chromosomes, whose evolution depends on mutation and crossover operations, as in the natural evolutionary process [5].

The basic GA steps are:

- Initialize random population.
- Evaluate chromosomes and check solution.
- Selection to crossover.
- Effect crossover.
- Mutation.

Evaluate the chromosomes and check solution, if the solution was not found then go back to step 2.

A. *Viral Infections in Genetic Algorithm*

The viral infection is not an innovation in GA. It was discussed other times like in VEGA [10] and GAVI [6]. In both methods is used another population composed by virus, called viral population and infection of chromosomes called transcription [6][10].

In VEGA the viral population is a subset of chromosomes, created from initial hosts [10].

During the process of infection, a virus is selected by rank process. However, as the virus has no fitness value, because it doesn't have a complete solution to be evaluated, it is used a parameter of infection, called *fitvirus* [10]. This is an indicator of how well the virus has acted. After selecting the virus that will effect the infection (in a particular chromosome), that time is made the transcription, which consists in the modification of the infected chromosome by the viral information that contains a section similar to the one represented by the infecting agent [6, 10].

When a virus is created or modified, its infectivity level is set at a fixed initial value. If the virus infects a chromosome (increase chromosome's fitness) their level of infectivity is increased by 1 (one), otherwise (if turn down the fitness) this value is reduced by 1 (one). If the virus infectivity value reaches 0 (zero) the virus discard it parts and copy a portion of chromosome to itself [6].

The main difference between VEGA and GAVI is because GAVI uses viral infection as operator, ignoring the operator of mutation, and VEGA is a complete GA [6, 10].

IV. Retroviral Iterative Genetic Algorithm

The main reason for the use of viral structure in the algorithm is the fact that these viruses are associated with a source of genetic innovation, which is influenced by the rapid rate of replication and changing [9].

For biological inspiration of RIGA, the family of retroviruses was chosen. These viruses do not possess correction mechanisms to undo possible genetic mutations that occur naturally during viral multiplication, which causes a high mutation rate, arising genetically modified individuals at each generation, what is considered an important characteristic during the processing time of GA [2, 8]. The acquisition of cellular genes was chosen as a method to viral evolution, since it is quite common in retroviruses [8].

There are so many differences between RIGA and GAVI and VEGA, some of them are:

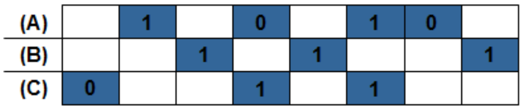
- RIGA doesn't change any GA component, GAVI remove the mutation operation.
- In the GAVI the worst viruses has them genetic material changed, in the RIGA they are completely changed, thereby, the viral population is constantly remade, increasing the possibility of infection.

- The biological basement of RIGA is very specific for the use of retroviral structure.
- VEGA creates virus only from host chromosomes, RIGA creates virus from host chromosomes and a new virus (providing high diversity of genetic material viral).
- VEGA and GAVI handle a virus as a sequenced subset of chromosome, in the other hand, RIGA handles virus with dispersed subset.
- The viral lifecycle and parameters in RIGA are well-defined.

In the next sessions are presented the components of RIGA.

A. Viruses

Viruses in RIGA are structures that have the same size of a chromosome, however, with some empty spaces, because the idea is to share genetic material and avoid other population of chromosomes working in parallel. The amount of empty spaces and its provisions are determined randomly. Thus for a problem that requires a binary representation of eight positions, some viruses have information as seen in Fig 2.

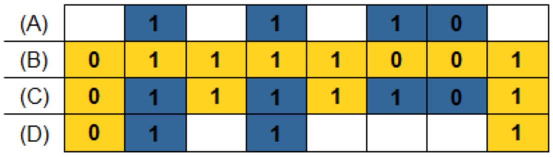


(A)		1		0		1	0	
(B)			1		1			1
(C)	0			1		1		

Fig.2. Possible viruses for a binary chromossome with eight positions.

B. Viral Population and Creation of New Viruses

For creation of the new viruses that will compose the viral population, RIGA was inspired by the natural process common in retroviruses called reverse transcription. The process consists basically of the following steps (Fig 3):

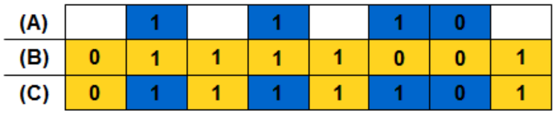


(A)		1		1		1	0	
(B)	0	1	1	1	1	0	0	1
(C)	0	1	1	1	1	1	0	1
(D)	0	1		1				1

Fig. 3. Creating a new virus process (A) Random virus (B) Chromossome from population (C) Auxiliar chromossome containing a mix of both genetic materials (D) New virus containing genetic material from virus (A) and chromossome (B).

C. Infection

Infection is the process of inclusion of the viral genetic material into the host chromosome, which is required a virus and a chromosome. The target chromosome will have changed their genetic material in the same positions where the genes are arranged on the virus, so all the viral genetic information, will be copied to the target chromosome, excepting the empty spaces, which will be filled by the host chromosome. The RIGA infection is represented in Fig 4.



(A)		1		1		1	0	
(B)	0	1	1	1	1	0	0	1
(C)	0	1	1	1	1	1	0	1

Fig. 4. Chromossome infection (A) Virus (B) Chromossome (C) Infected Chromossome.

The infection process depends exclusively on one single factor: increasing of chromosome fitness. The infection is successful when an infected chromosome has an increase on its fitness and unsuccessful when it has a decrease on its fitness. This is an important factor because it determines which viruses will infect the next generation. The worst viruses (with less infection) will be extinct. The reason of the infectious process is restrict to the increase in fitness, because if was considered any kind infection, good chromosomes could become bad. Thus, only the successful infections are important to RIGA.

D. Parameters

The RIGA uses the same parameters from classic GA (number of individuals, rate of mutation, crossover and elitism and the type of selection and crossover). However, to apply the concepts of viral infection by retroviruses, the defined parameters are:

1. Infection population rate: the rate of chromosses that will be infected in generation. This parameter will cause a slow execution if the value is higher than 50.
2. Viral elitism rate: the rate of viruses that will be kept in the next generation of viral population, if the value is more than 50, the evolution is compromized.
3. Number of viruses: the number of viruses of viral population, the higher that number, the lower is the time of execution.
4. Weakest infection: this parameter forces the infection of weakest chromossome, cause a fast execution, but results in more generations.
5. Single infection: this parameter forces a unique infection per chromossome, cause a fast execution, but results in more generations.
6. Internal infection rate: this parameter indicates the maximum percentual of genetical material from any chromossome that will form a new virus.

E. The Algorithm

RIGA has an additional step in the traditional algorithm (step 6), above:

- Initialize random population.
- Evaluate chromosomes and check solution.
- Selection to crossover.
- Effects crossover.
- Mutation.
- Viral application
  - If it is the first time, generate a random viral population.
  - Generate new virus based in chromosome population.

- Infect chromosomes.
  - Infect each randomly chosen chromosome with each existent virus.
  - For each successful infection, increment 1 to virus, however, decrement 1.
  - Check the virus with highest infection rate and keep according with viral elitism rate.

– Evaluate the chromosomes and check solution, if the solution wasn't found then go back to step 2.

V. Results

To the test of RIGA, the implementation was based to support the functions: F1 (Shifted Sphere Function), F2 (Shifted Schwefel's Problem), F3 (Shifted Rotated High Conditioned Elliptic Function) and F5 (Schwefel's Problem 2.6 with Global Optimum on Bounds) [7].

Considering all experiments performed (a total of 80 executions for 40 using RIGA and 40 using GA with same configurations per function and initial population), the RIGA was superior to GA in all the experiments, getting closer to the optimal result in all of them (Fig. 5 to Fig. 8).

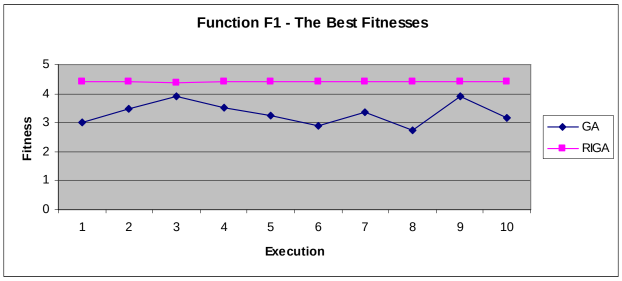


Fig. 5. The best fitnesses for function F1.

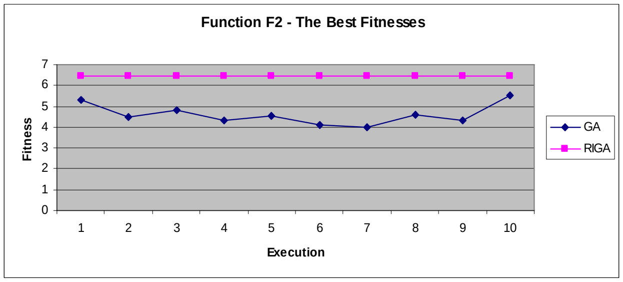


Fig.6. The best fitnesses for function F2.

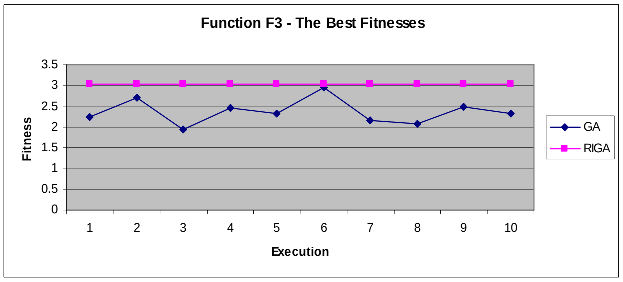


Fig. 7. The best fitnesses for function F3.

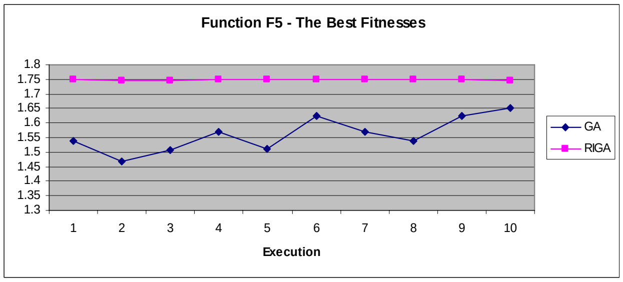


Fig. 8. The best fitnesses for function F5.

According to Table 1, it is possible verify that the average fitness of the RIGA was even higher (F1, F2, F3 and F5) than the best fitnesses found in some executions by the GA. Thus, we can affirm that the RIGA, with the same population, elevates the fitness of individuals considered bad, getting closer to an optimal result in only 100 generations generated by RIGA which were lower in all cases.

TABLE I
THE BEST EXECUTIONS FOR FUNCTIONS F1, F2, F3AND F5

CEC-LIB		GA		RIGA	
F(x)	*Max(F(x))*	*BF*	*AF*	*BF*	*AF*
F1	$4{,}4 \times 10^4$	**3,9214**	1,0015	**4,3893**	3,9341
F2	$6{,}5 \times 10^4$	**5,4998**	1,4292	**6,4784**	5,6315
F3	$3{,}8 \times 10^{10}$	**2.9530**	0.5893	**3.0421**	2.7696
F5	$1{,}8 \times 10^4$	**1.6513**	0.6740	**1.7486**	1.6474

(BF) Best Fitness (AF) Average Fitness

Thereby, it is correct to affirm that the method (RIGA) was more efficient for mathematical problems exposed.

VI. Conclusions

This work presented the creation of a genetic algorithm based on the structure of the retroviruses family. The biological application in conjunction with genetic algorithm took this algorithm to be called Retroviral Iterative Genetic Algorithm.

After the present work, it was found that the retroviral biological inspiration had a beneficial effect in the application of RIGA compared to the GA, since all the RIGA results were significantly better when compared to GA results.

Thus, with positive results obtained by RIGA, we conclude that RIGA proves to be more efficient when compared to GA for the functions exposed in the work. Still, this should not be taken as a rule for any problem involving resolutions of GA's problems.

References

[1] J. Carter and V. Saunders, *Virology Principles and Applications*. John Wiley & Sons ltd., England, 2007.

[2] R. L. Haupt and S. E. Haupt, *Practical Genetic Algorithms*. John Wiley & Sons ltd., England, 1998.

[3] S. Hogg, *Essential Microbiology*. John Wiley & Sons ltd., England, 2005.

[4] A. Agut, "Um Sistema Estratégico De Reprodução," *Scientific American Brasil*, Edição Especial, N. 28, p. 14-19, São Paulo, 2009

[5] R. Linden. *Algoritmos Genéticos 1*. Ed. Rio De Janeiro: Brasport 2006.

[6] A. Guedes, J. Leite, D. Aloise, “Um Algoritmo Genético Com Infecção Viral Para O Problema Do Caixeiro Viajante,” *Revista PublICa*,Ano IV, vol 4, 2005.

[7] P. Suganthan, N. Hansen, J. Liang, K. Deb, Y. Chen, A. Auger, S. Tiwari, “Problem Definitions and Evaluation Criteria For The Cec 2005 Special Session On Real-Parameter Optimization,” *Technical Report*, Nanyang Technological University, Singapore and Kangal Report Number 2005005 (Kanpur Genetic Algorithms Laboratory, Iit Kanpur), 2005.

[8] M. Mitchell, *An Introduction to Genetic Algorithms*. MIT Press, 1999.

[9] L. Villarreal, “Virus São Seres Vivos?” *Scientific American Brasil*, Edição Especial, n. 28, p. 21-24. São Paulo, 2009.

[10] N. Kubota and T. Fukuda and K. Shimojima, “Virus-evolutionary genetic algorithm for a self-organizing manufacturing system,” *Computers & Industrial Engineering*, vol 30, pp. 1015-1026, 1996.

Expected Utility from Multinomial Second-order Probability Distributions

David Sundgren

Abstract—**We consider the problem of maximizing expected utility when utilities and probabilities are given by discrete probability distributions so that expected utility is a discrete stochastic variable. As for discrete second-order distributions, that is probability distributions where the variables are themselves probabilities, the multinomial family is a reasonable choice at least if first-order probabilities are interpreted as relative frequencies. We suggest a decision rule that reflects the uncertainty present in distribution-based probabilities and utilities and we show an example of this rule in action with multinomial second-order distributions.**

Index Terms—**Imprecise probability. second-order probability, discrete probability distributions, multinomial distributions, expected utilty.**

I. Introduction

WHEN computing the expected utility of a decision alternative it may not always be possible to give precise values for the utilities and probabilities of the possible outcomes. The model for imprecise probabilities used here is discrete second-order probability distributions. The term second-order probability comes from the notion that the distributions express the probability that a first-order probability has a certain value. As motivation for discrete second-order distributions we may consider updating, the form of available information and computation of expected utility.

In a continuous second-order setting, a lower bound of a probability can rarely if ever be the result of an observation. But after seeing a three-eyed dog in a kennel of ten, I know that at least one out of ten dogs in that kennel has three eyes. Outside the kennel, I cannot, based on the observation, say much more than that the probability of coming across a three-eyed dog is non-zero. Thus, in situations where the available data has the form of relative frequencies, discrete rather than continuous second-order distributions would be a suitable choice for describing imprecise probabilities. In the case of subjective probabilities, one might as well use discrete distributions unless one has the need to express the probability of particular irrational probability values. For instance that I believe that the probability of seeing another three-eyed dog in my life-time is at least $1/\pi$. That is, discrete second-order distributions are suitable for both objective and subjective probabilities while continuous distributions come into its own in subjective settings. As for computation of distribution expected utility, the fact that there are a finite number of points in a discrete distribution makes a direct computation possible. In the continuous case simulations are necessary.

There is a rich literature on imprecise probabilities, see e.g. [1], [2], [3], [4], [5]. Here second-order probabilities, see e.g. [6], [7], [8], [9], in general and discrete second-order distributions in particular are used and advocated as opposed to interval based models. The standard interval based approach to imprecise probabilities is to employ sets of probability measures, also called credal sets. A credal set is informally a set of probability distributions. Such a set is usually restricted by lower (or, equivalently, upper) bounds of probabilities, and the demand that the set is convex. The intuition appears to be that instead of choosing precise probabilities one uses the set of all probability distributions that are consistent with the beliefs of an agent. Theories of this kind include Choquet capacities [10], lower probabilities [11] and lower previsions [12]. These theories are accessibly summed up in [13]. Without going into the details of these advanced theories one might with an extreme simplification say that in the traditional models of imprecise probabilities the decision maker's or expert's knowledge is represented by intervals between the lowest and highest possible values of the probabilities.

But given some form of representation of imprecise probabilities and utilities it is often not evident what decision rule to employ. With interval based models of imprecise probabilities there are, if we choose to maximize a utility rather than minimize a cost, the rules of Γ-maximax [14], Γ-maximin [15], E-admissibility [16], [17], maximality [12] and interval dominance. [18], see [19] for a comparison of these decision rules. These rules have in common that they single out one or several decision alternatives as optimal without qualifying the ranking with regard to the uncertainty inherent in imprecise probabilities.

In contrast, with utilities and probabilities that are expressed by belief or probability distributions, second-order probabilities, one can measure the imprecision or uncertainty with e.g. variance. With this in view it would be reasonable to use a decision rule that reflects the amount of uncertainty. For example, if the probabilities for all possible utility and (first-order) probability values are known, one can compute the probability that one alternative gives higher expected utility than another alternative.

With continuous second-order probability distributions it seems to be difficult to find closed expressions for expected utility, see [20]. In practice, simulations would have to be made. Alternatively, a decision maker could use discrete distributions. Albeit closed expressions for expected utility would still be hard to find, the expected utility values can be easily computed when second-order distributions are given. Here multinomial distributions are used, but the point is the use of discrete second-order distributions. Discrete distributions offer an environment for updating that is hard to conceive of with continuous second-order distributions and at least brute-force computation of distribution of expected utility is more straight-forward. The multinomial distribution family is used here as an example because of its simplicity, future research will undoubtedly reveal distributions with more attractive properties.

II. Multinomial Second-order Distributions

Given that the variables of a second-order distribution are themselves probabilities, there is a normalization constraint in that probabilities must sum to one. For this reason it is hard for a decision maker to construct a second-order distribution by looking at the possible outcomes separately, and to consider all outcomes simultaneously might be untenable.

Manuscript received June 8, 2010. Manuscript accepted for publication July 25, 2010.

D. Sundgren is with the University of Gävle, Sweden (dsn@hig.se).

But the decision problem itself might imply principles that restrict the choice of second-order distributions. E.g. if it is possible to look at the probabilities of the possible outcomes as relative frequencies, and if the knowledge that restricts the lower bounds of the probabilities come from observations, multinomial distributions are natural candidates, as we shall see.

Consider an experiment with $N = \sum_{i=1}^n k_i$ objects of n different types. Assume that the probability to pick an object of type i is $1/n$. Since there are $\frac{N!}{\prod_{i=1}^n k_i!}$ permutations of the N objects, if there are k_i objects of type i, the probability that there k_i objects of type $i, i = 1, \ldots, n$ is

$$\Pr(k_1, k_2, \ldots, k_n) = \frac{N!}{n^N \prod_{i=1}^n k_i!}$$

The marginal distribution for a single number of objects is

$$\Pr(k) = \sum_{k_i \neq i} \Pr(k_1, k_2, \ldots, k_n) = \frac{(n-1)^{N-k}}{n^N} \binom{N}{k},$$

which is also the probability that there are k objects of a certain type among a total of N objects.

If we gain information by looking at a few of the objects and observe a_i objects of type i, the uncertainty is reduced to the remaining $N - \sum_{i=1}^n a_i$ objects of unknown type. So the question is how many more than the observed a_i objects there are of type i. The updated probability should be

$$\frac{(N - \sum_{i=1}^n a_i)!}{n^{N - \sum_{i=1}^n a_i} \prod_{i=1}^n (k_i - a_i)!}$$

But updating the prior $\frac{N!}{n^N \prod_{i=1}^n k_i!}$ with the hypergeometric likelihood

$$\Pr(a_i|k_i) = \frac{\prod_{i=1}^n \binom{k_i}{a_i}}{\binom{N}{\sum_{i=1}^n a_i}}$$

gives just the posterior distribution $\frac{(N - \sum_{i=1}^n a_i)!}{n^{N - \sum_{i=1}^n a_i} \prod_{i=1}^n (k_i - a_i)!}$ suggested above. The likelihood $\Pr(a_i|k_i)$ is the probability that one can see a_i things among the $\sum_{i=1}^n a_i$ observed given that there are in total k_i objects of type i.

What possibly is new here is that since we consider the frequencies k_i/N as probabilities, the multinomial distributions described here are examples of discrete second-order distributions, that is, describing the probabilities of different probability values. The advantages of discrete second-order distributions would include Bayesian updating as described above.

We must note, though, that the choice $1/n$ of underlying probabilities for the n types of objects is rather arbitrary and even questionable. For instance, if a relatively large number a_i objects of type i have been observed, it might be that the probability is larger than $1/n$. Work on more sound, alternatively less arbitrary, discrete second-order distributions is under way. For the purpose of this paper, though the multinomial distributions here are sufficient. We wish to show that known discrete probability distributions can serve as second-order distributions, to see how updating might work and suggest a decision rule based on discrete second-order probabilities. The particular distribution family used here is just an example, and the decision rule suggested here and the associated algorithms do not depend on any particular form of discrete second-order probability distribution.

TABLE I
Utility Distributions for Alternative A

Outcome	0	1	2	3	4	5	6	7	8
1	.3	.3	.2	.2	0	0	0	0	0
2	0	0	0	.3	.4	.3	0	0	0
3	0	.1	.2	.2	.2	.2	.1	0	0
4	0	0	0	0	.1	.4	.3	.2	0

We conclude this section with a remark on upper bounds of probabilities. the lower bounds are given by a_i/N, meaning that a_i objects have been observed. But then there can be at most $N - \sum_{j \neq i} a_j$ items of type i, so the upper bound of each probability is given by the lower bounds of the other probabilities.

For instance, let $N = 8$ and $n = 4$. Further, let the lower bounds be $p_1 \geq 0, p_2 \geq 1/8, p_3 \geq 3/8$ and $p_4 \geq 0$. Then we know that $p_1 \leq 1 - 1/8 - 3/8 = 1/2, p_2 \leq 1 - 3/8 = 5/8, p_3 \leq 1 - 1/8 = 7/8$ and $p_4 \leq 1 - 1/8 - 3/8 = 1/2$.

The corresponding multinomial second-order distribution is

$$\Pr(k_1/8, k_2/8, k_3/8) = \frac{4!}{k_1!(k_2-1)!(k_3-3)!(8-k_1-k_2-k_3)!4^4}$$

Distribution of expected utility is a crucial part in the decision rule suggested here, and in the example below the distribution will be computed in a few cases. If second-order probabilities are given by the multivariate distribution $p(k_1, k_2, \ldots, k_n)$ where $\sum_{i=1}^n k_i = N$ (i.e. the first-order probabilities are the ratios k_i/N), and we have n independent utilities given by distributions $p_i(u_i)$, the probability mass function of expected utility is

$$h(z) = \sum_{\sum_{i=1}^n k_i u_i = z} p(k_1, k, \ldots, k_n) p_i(u_i)$$

For mathematical reasons concerning primality and co-primality it is hard to make general claims as to which values of k_i and u_i that are involved for every expected utility value z. Thus a brute-force technique will be employed in this paper.

III. A Decision Problem with Four Alternatives

Let us look at the four constructed decision alternatives A, B, C or D. Below we give the distributions for the utilities, and the lower bounds for probabilities that serve as parameters for our multinomial distributions. The probability and utility values are arbitrarily chosen for the sake of the example, they are not derived from any observations or any assessments of real situations. On the other hand, the model employed in the example can be used regardless of how data is collected, be it from observations or by subjective judgments.

A. Alternative A

In the first alternative there are four possible outcomes. The utilities of these are given by the distributions in table I.

The probabilities have lower bounds $p_1 \geq 0, p_2 \geq 1/8, p_3 \geq 3/8$ and $p_4 \geq 0$, so the multinomial second-order distribution is

$$\Pr(k_1/8, k_2/8, k_3/8) = \frac{4!}{k_1!(k_2-1)!(k_3-3)!(8-k_1-k_2-k_3)!4^4},$$

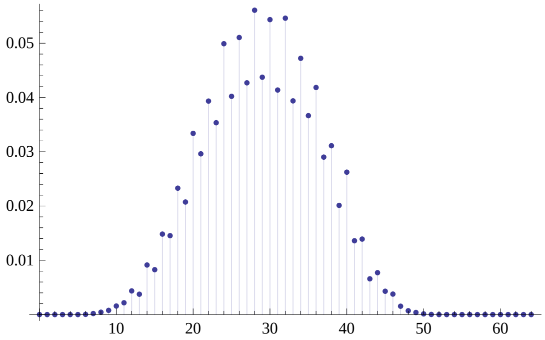


Fig. 1. Expected utility of alternative A

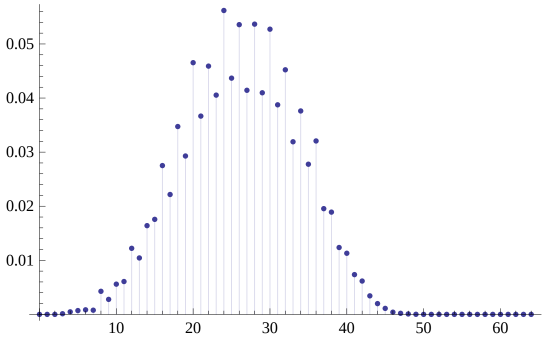


Fig. 2. Expected utility of alternative B

as in the example above at the end of Section II. These distributions for utilities and probabilities give the expected utility distributed as in Figure 1 below.

B. Alternative B

For alternative B the utilities are the same as for alternative A, the only difference between alternatives A and B is that the probability of the first outcome is higher at the expense of the probability of the second outcome; $p_1 \geq 1/8, p_2 \geq 0, p_3 \geq 3/8, p_4 \geq 0$ and the second-order probability distributions is

$$\Pr(k_1/8, k_2/8, k_3/8) = \frac{4!}{(k_1-1)!k_2!(k_3-3)!(8-k_1-k_{-}k_3)!4^4}$$

The resulting distribution of expected utility is plotted in Figure 2.

C. Alternative C

Alternative C is distinguished in that one of the possible outcomes has two possible sub-outcomes with probabilities $p_{11} \geq 1/4, p_{12} \geq 0$. The utilities of the sub-outcomes are found in Table II.

Outcome 1 then has a distribution of expected utility as plotted in Figure 3.

TABLE II
UTILITY DISTRIBUTIONS FOR OUTCOME 1 OF ALTERNATIVE C

Sub-outcome	0	1	2	3	4	5	6	7	8
1	.2	.3	.3	.2	0	0	0	0	0
2	0	0	.2	.4	.2	.2	0	0	0

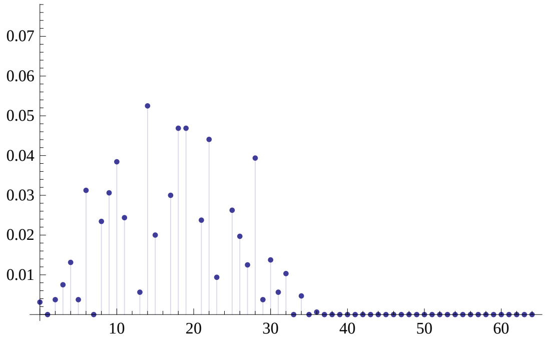


Fig. 3. Expected utility of outcome 1 in alternative C

TABLE III
UTILITY DISTRIBUTIONS FOR ALTERNATIVE C

Sub - outcome	0	1	2	3	4	5	6	7	8
1	.063	.255	.382	.262	.038	.001	0	0	0
2	0	0	.2	.5	.3	0	0	0	0
3	0	0	0	.1	.2	.3	.2	.1	.1

This expected utility distributions in turn serves as utility distribution for outcome 1. But in order to facilitate computation of expected utility of alternative C, the 65 values in Figure 3 are collected into 9 values corresponding to the scale 0 – 8 employed for the other utilities. See Table III.

The probabilities for the three main outcomes are $p_1, p_2 \geq 1/8$ and $p_3 \geq 3/8$, so the multinomial second-order distribution is

$$\Pr(k_1/8, k_2/8) = \frac{3!}{(k_1-1)!(k_2-1)!(5-k_1-k_2)!3^3} ,$$

resulting in the expected utility distribution plotted in figure 4.

D. Alternative D

As with alternative C there are three possible outcomes in alternative D but no sub-outcomes.

And since the lower bounds of probabilities are $p_1 \geq 0, p_2 \geq 1/8, p_3 \geq 1/8$ the second-order distribution is

$$\Pr(k_1/8, k_2/8) = \frac{6!}{k_1!(k_2-1)!(7-k_1-k_2)!3^6}$$

A plot of the distribution of expected utility is found in Figure 5

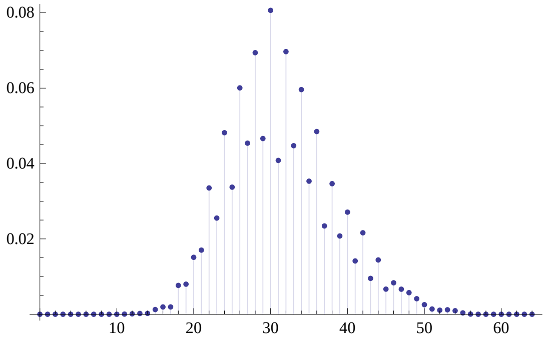


Fig. 4. Distribution of expected utility of alternative C

TABLE IV
UTILITY DISTRIBUTIONS FOR ALTERNATIVE D

Outcome	0	1	2	3	4	5	6	7	8
1	.1	.1	.2	.2	.2	.1	.1	0	0
2	.1	.1	.1	.1	.1	.2	.1	.1	.1
3	.1	.1	.1	.2	.2	.1	.1	.1	0

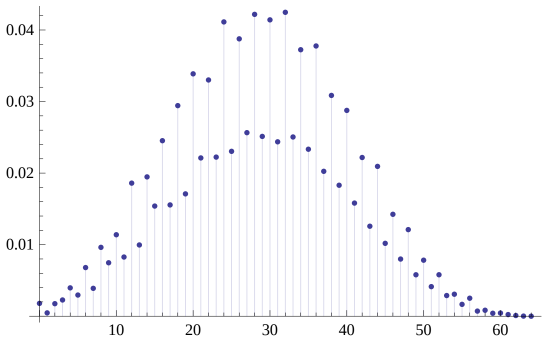


Fig. 5. Distribution of expected utility of alternative D

E. Maximizing Expected Utility

We can begin by ranking he alternatives by expectation of expected utility, $\sum_{z=0}^{64} zh(z)$, we then see that

$$\sum_{z=0}^{64} zh_A(z) = 28.9, \sum_{z=0}^{64} zh_B(z) = 26.2,$$
$$\sum_{z=0}^{64} zh_C(z) = 31.3, \sum_{z=0}^{64} zh_D(z) = 28.8\,,$$

so the ranking is C, A, D, B. Obviously, there is little difference between alternatives A and D, but what is the probability that A yields higher expected utility than D? Or that the highest ranking alternative C gives a better result than the second best A?

With probability

$$\sum_{z=0}^{64} \sum_{x=0}^{z} h_C(z) h_A(x) = .6046$$

alternative C would give at least as high expected utility as alternative A. Given that anything less than .5 would mean that C is worse than A, the superiority of C over A is not very impressive. The point is that the uncertainties inherent in the second-order probabilities and utilities carry over to uncertainty of expected utility and that given the probability of 60 % one can give a cautious recommendation for alternative C.

Further, the probability that A gives at least as high expected utility as D is a mere .52, as slight change in the underlying data could reverse the verdict in favor of D. And with a probability of .59 alternative D is at least as good as B.

The supports for distributions of expected utility, or interval-based expected utilities, are as follows,

$$6 \leq \mathrm{EU}_A \leq 51, 3 \leq \mathrm{EU}_B \leq 49, 11 \leq \mathrm{EU}_C \leq 57, 0 \leq \mathrm{EU}_D \leq 63$$

Overlapping of intervals makes it hard to draw conclusions in terms of which is the better alternative without applying a maximax or maximin rule. The highest lower bound of expected utility is 11 for alternative C but the highest upper bound is 63 for D. In such an interval-based decision analysis there would not be room for estimation of how probable these extreme values are.

IV. TIME COMPLEXITY

The relevant parameters for the complexity of computing discrete expected utility are n, the number of possible outcomes, and N, the number of points in the discrete distributions. The number of points are not necessarily the same for probabilities and utilities, or even between probabilities or utilities, but for simplicity we assume that they are, as in the example above. In any case the different numbers of distributions points would hardly differ by magnitudes.

Assuming $N+1$ first-order probability values k_i/N ranging from 0 to 1 and $N+1$ utility values, there are N^1+1 different possible values of expected utility, from 0 to N^2. Given expected utility z, we must collect all allowable probability and utility vectors that produce the value z of expected utility. But since there is not as yet an expression for the minimal solution x to a diophantine equation $ax+by=c$, (the formula $x = a^{-1}c(\mathrm{mod} b)$ depends on a and b being co-prime), we cannot use a closed expression for directly computing the distribution of discrete expected utility. The raggedness of the plots give some indication that discrete distributions of expected utility might not have simple expressions independent of primality and co-primality. Instead we have to go through all $(N+1)^{2n-1}$ possible choices of $n-1$ probabilities and n utilities and see whether they produce the expected utility z. Each such check costs $\mathcal{O}(\log^2 N)$ arithmetic operations. In total we have $\mathcal{O}(N^{2n} \log^2 N)$ operations for computing expected utility.

If we consider the number of possible consequences n of a decision alternative as a constant inherent in the problem, the time complexity is polynomial in N, meaning that increasing the granularity of the distributions is not prohibitively expensive. However, it is also possible to imagine that deeper investigation of the decision problem leads to splitting of consequences, thus increasing n. Than again, such a situation would rather lead to deeper levels of the decision tree as in alternative B as in our example in Section III. Then n does not change, but computing the distribution of utility for the sub-events makes for another $\mathcal{O}(N'^{2n'} \log N')$ operations.

We have suggested a decision rule based on expectation of expected utility and probability assessments of the probability that one alternative yields higher expected utility than another.

Expectation of expected utility is computed by N^2+1 multiplications and N^2 additions, $\mathcal{O}(N^2 \log N)$. And the suggested comparison of alternatives, the probability that, say alternative A has at least as high expected utility as alternative B, $\sum_{z=0}^{N^2} \sum_{x=0} zh_A(z) h_B(x)$ means

$$\frac{N^2(N^2+1)(2n^2+1)}{12} - \frac{N^2(N-1)}{4} \in \mathcal{O}(N^6)$$

multiplications, or $\mathcal{O}(N^6 \log^2 N)$ elementary operations.

V. CONCLUSIONS

With second-order probabilities it is possible to express any consistent beliefs about the probabilities of an event. In fact, it is not hard to imagine that a decision maker might shy away from all the possibilities and express his or her knowledge through e.g. intervals. but it may be that the nature of the decision problem makes some second-order distributions more suitable than others. Such differentiation will be a matter for future research. Also, given a certain distribution family, consistency constraints might limit the choice to the lower bounds of the probabilities. Indeed such local information might be all that the decision maker has access to. Here we have looked at multinomial distributions for the purpose of expressing second-order probabilities.

It has been demonstrated that discrete second-order probability distributions allow for updating through observations in a way that continuous distributions would not. And such discrete distributions lend themselves naturally to probability viewed as relative frequency. Furthermore, even when relative frequencies are inappropriate or unavailable, or simply when subjective probabilities are desired or required, continuous second-order distributions offer more possibilities than their discrete counterparts only in rather contrived examples.

In the second-order model, uncertainty is in a manner of speaking made precise. For utilities, probabilities and expected utility, variances may be computed, or the probability that the probability of an outcome is lower than a given value, or the probability that a decision alternative has a higher expected utility than another alternative. Such computations are however for the foreseeable future impossible to conduct save by simulation when using continuous distributions.

We have shown an example of second-order decision making with discrete distributions and shown that the necessary calculations are computationally costly, but far from intractable.

References

[1] A. P. Dempster, "Upper and lower probabilities induced by a multivalued mapping," *Annals of Mathematical Statistics, vol xxxviii*, pp. 325–399, 1967.

[2] G. Shafer, *A Mathematical theory of evidence*. Princeton University Press, 1976.

[3] P. Huber, "The case of choquet capacities in statistics," *Bulletin of the International Statistical Institute, 45*, pp. 181–188, 1973.

[4] P. Huber and V. Strassen, "Minimax tests and the neyman-pearsons lemma for capacities," *Annals of Statistics, 1*, pp. 251–263, 1973.

[5] I. Good, "Subjective probability as the measure of a non-measurable set," in *Logic, Methodology, and the Philosophy of Science*, P. Suppes, B. Nagel, and A. Tarski, Eds. Stanford University Press, 1962, pp. 319–329.

[6] R. F. Nau, "Uncertainty aversion with second-order utilities and probabilities," *Management Science*, vol. 52, no. 1, pp. 136–145, 2006.

[7] L. Ekenberg and J. Thorbiörnson, "Second-order decision analysis," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 9, No 1*, vol. 9, no. 1, pp. 13–38, 2001.

[8] L. V. Utkin and T. Augustin, "Decision making with imprecise second-order probabilities," in *ISIPTA '03 - Proceedings of the Third International Symposium on Imprecise Probabilities and Their Applications*, 2003, pp. 547–561.

[9] L. V. Utkin, "Imprecise second-order hierarchical uncertainty model," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, Volume 11:3*, pp. 301–317, 2003.

[10] G. Choquet, "Theory of capacities," *Ann. Inst. Fourier*, pp. 5:131–295, 1954.

[11] C. A. B. Smith, "Consistency in statistical inference and decision," *Journal of the Royal Statistical Society, Series B, xxiii*, pp. 1–25, 1961.

[12] P. Walley, *Statistical reasoning with Imprecise Probabilities*. Chapman and Hall, 1991.

[13] F. Cozman, "A brief introduction to the theory of sets of probability measures," cMU Tech. report CMU-RI-TR 97-24.

[14] J. K. Satia and R. E. L. Jr., "Markovian decision processes with uncertain transition probabilities," *Operations Research*, vol. 21, pp. 728–740, 1973.

[15] H. Robbins, "Asymptotically sub-minimax solutions to compound statistical decision problems," *Proc. Second Berkerly Symp. Math. Stat. Probab.*, vol. 1, 1951.

[16] I. J. Good, "Rational decisions," *Journal of the Royal Statistical Society, Series B*, vol. 14 (1), pp. 107–114, 1952.

[17] I. Levi, *The Enterprise of Knowledge*. MIT Press, 1980.

[18] M. Zaffalon, K. Wesnes, and O. Petrini, "Reliable diagnoses of dementia by the naive credal classifier inferred from incomplete cognitive data," *Artificial Intelligence in Medicine*, vol. 29, no. 1-2, pp. 61–79, 2003.

[19] M. C. Troffaes, "Decision making under uncertainty using imprecise probabilities," *International Journal of Approximate Reasoning*, vol. 45, no. 1, pp. 17 – 29, 2007.

[20] D. Sundgren, L. Ekenberg, and M. Danielson, "Some properties of aggregated distributions over expected values," in *MICAI 2008: Advances in Artificial Intelligence*, ser. Lecture Notes in Computer Science, no. 5317. Springer, 2008, pp. 699–709.

Analysis of Multipath Propagation based on Cluster Channel Modelling Approach

Marvin R. Arias

***Abstract*—The computer simulation approach with an emphasis on the propagation modelling for wireless channels for current and future communication systems is a powerful tool to asses the performance of systems without the need of building them. This paper presents a clustering approach geometry-based channel model, and employs it to derive the power density function (PDF) of the Angle of Arrival (AOA) of the multipath signal components. To evaluate the theoretical clusters PDF in angular domain proposed, we make computer simulations for the geometry-based channel model proposed and compared it with experimental results published in the literature showing good agreement. The clusters PDF derived can be used to simulate the power-delay-angle profile (PDAP) and to quantify second order statistics, i.e., power angular spectrums (PAS) and the associated angular spreads (Ass) for a given elliptical shape of the cluster.**

***Index terms*—Angle-of-arrival, angular spread, antenna arrays, channel modelling, clustering, multipath propagation.**

I. Introduction

RADIO wave propagation in the urban environment is an important issue in the study of wireless communication. In the study of channel propagation, various models, such as empirical models and stochastic modelling based on geometry [1]-[5], is commonly used. In third generation systems like the wideband Code Division Multiple Access (W-CDMA), the physical channel is characterized by multipath propagation. In a scattering environment the propagation paths to a receiving antenna will come from a certain angular spread

Of directions; typically, several versions of the transmitted signal impinge on the receiving antenna from different directions because of multipath. In fact, antenna arrays can be used to implement space-time selective transmission in the downlink and to provide radio localization services. To do so, however, separate knowledge of the directions from which the signals arrive, or angle-of-arrival (AOA), is an important property when characterizing the channel as well as designing receiver algorithms [2]. Since the time and more dominantly the angular spread of the multipath components greatly determines the performance of wireless communication systems. The angular spread essentially determines the diversity gain by using an antenna array [3]. Employing antenna arrays has also been proposed to reduce the co-channel interference by transmitting energy only in the direction of a specific user and essentially no energy in the directions of other users. The spread is important for diversity schemes and also for determining the AOA. Thus, characterizing the channel in AOA terms is an interesting alternative to standard models. It is now an accepted model that an angular spread occurs from a cluster of scatterers; where the total signal may come from several clusters, see [4]-[8] and referenced there in. As mentioned in [9] a statistical approach is necessary to understand the basic mechanism of propagation.

Several measurement campaigns done in different indoor and outdoor environments, report that multipath components (MPCs) arrive in clusters. Each cluster consists of a group of MPCs with similar angles of arrival (AOAs), angles of departure (AODs), and time of arrival (TOA), corresponding to a dominant path at the receiver (Rx), see [4]-[6], and [9]. Measurements results reported in [4]-[6], and [9], show that the Palladian function is the best fit for estimated AOA in outdoor environments In this paper, essentially, the main objective is to analytically derive the power density function (PDF) of AOA of the multipath signal for geometry-based statistical channel model that is valid for a circular and elliptical scattering, we assume the double bounce approach channel model as described in [7], [10]. Our contribution would be useful in simulating the power-delay-angle profile (PDAP) and to quantify second order statistics, i.e., angle spread for a given elliptical shape of the cluster of the multipath signal operating in a variety of propagation conditions in urban environment. The propagation environment is composed of scatterers, which are grouped in clusters. Among others, the number of clusters and the average number of scatterers within a cluster can be set with a parameter.

The remainder of the paper is organized as follows. The next section describes the proposed clustering approach channel model and in section 3 based on the channel model proposed, we derive the marginal PDF of the AOA for urban environments. Section 4 presents the simulation results and comparison with experimental results for one typical outdoor scenario in urban environment. Finally, section 5 contains some concluding remarks.

II. Clustering Approach Channel Model

This section briefly discussed the clustering approach geometry-based channel model that we use to derive the marginal PDF of the AOA for uniform scatterer density function. We assume that the mobile station (Rx) is stationary or at very low speed and therefore ignore the Doppler effects

Manuscript received May 3, 2010. Manuscript accepted for publication September 10, 2010.

The author is with Department of Electrical Engineering, National University of Engineering, P.O. Box 5595, Managua, Nicaragua (marao@ibw.com.ni).

in the analysis. Since the channel model is geometry-based; (as illustrated in Fig. 1), the signal statistics depend on the position of the base station (Tx) and mobile station (Rx), and the geometrical distribution of the clusters.

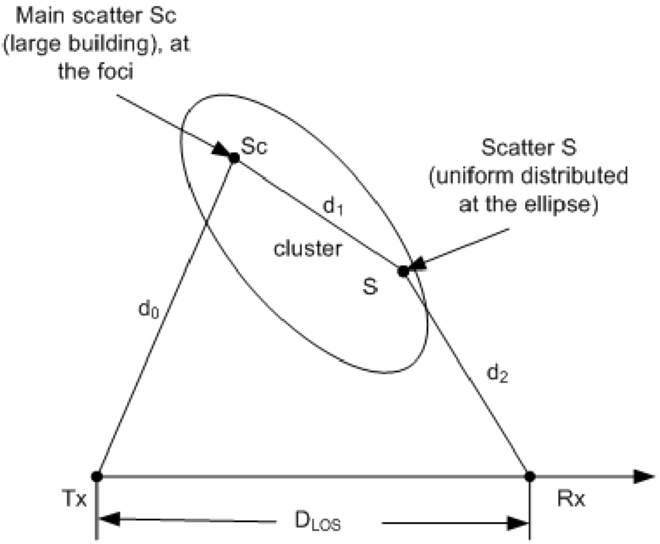


Fig. 1. Geometry-Based clustered multipath propagation model description.

We assume that each cluster (which includes a number of scatterers) is stationary in the far field. We define the angle-delay resolved impulse response of each cluster as [7]

$$h_{(cluster)j}(\tau,\phi)=\sum_{j=1}^{J}\sum_{i=0}^{L_j}\alpha_{ij}\delta(\phi-\phi_{ij})\delta(\tau-\tau_0-\tau_{ij}), \qquad (1)$$

where for each cluster: $|\alpha_{ij}|$ is the magnitude of the scattered source, ϕ_{ij} is the angle-of-arrival of the ith multipath component in the jth source, τ_0 represents the extra delay due to the double bounce effect and τ_{ij} is the delay associated with that component in the jth source. The parameter L_j is the total number of multipath components associated with the jth source. We define the total baseband channel impulse response as follows:

$$h_{btotal}(\tau,\phi)=\sum_{j=1}^{J}h_{(cluster)j}(\tau,\phi), \qquad (2)$$

The parameter J is the number of clusters. We analyze for the case of J=1, i.e., for one cluster condition. From the measurements reported in [4], [6],and [9] the receiver sees six clusters at most in 90% of the cases, thus we use J as a reference for the parameter, i.e. J ≤ L.

We assume the following restrictions: the cluster region has an elliptical shape because anything outside the ellipse has a large excess delay, i.e., the physical interpretation is that only multipath components with time delay smaller than the specified maximum time delay (bounded by the ellipse), are considered. Therefore, providing that the maximum delay is sufficiently large, nearly all of the power of the multipath signals of a physical channel will be accounted for by the model. The main scatterer (Sc), which is a large obstacle far away from the receiver, (e.g. a large building), is situated at one of the foci of the ellipse as shown in Fig.1 Moreover, there is line of sight (LOS) between the main scatterer and the Rx, and all scatterers (S) belonging to the same cluster are uniformly distributed inside it. We further assume that the propagation takes place in the horizontal plane containing the Rx, the Tx and the cluster, which are placed in the same plane. D_{LOS} represents the Tx-Rx separation distance. In the next section we analyse the approach to derive the marginal PDF of the AOA, using the clustering approach model proposed.

III. Cluster PDF of Angle of Arrival

Here we present an approach to derivate simple general formulas to model the marginal power density function (PDF) of the AOA between the receiver (*Rx*) and the far clusters. Assuming an elliptical shape of the clusters bounded by a circular shape, the approach is described as follows: Firstly in Fig. 2, we describe the details regarding the AOA of the cluster's multipath components (MPCs) at the receiver (*Rx*).

We express the scatterer's density function with respect to the polar coordinates *(r, θ)*, which are related to the rectangular coordinates via the following set of equations:

$$\begin{aligned} r&=\sqrt{x^2+y^2}, \\ \theta&=\arctan\left(\frac{y}{x}\right), \\ x&=r\cos(\theta), \\ y&=r\sin(\theta), \end{aligned} \qquad (3)$$

Where (*x, y*) denotes the position of the cluster. As shown in Fig. 2, the maximum AOA for the circular case is given by

$$\varphi_{c\max}=\arcsin\left(\frac{a}{R_c}\right), \qquad (4)$$

Where *"a"* is the radius of the circle and R_c is the distance between the centre of the circle and the *Rx*. Squeezing in the vertical dimension by a factor $r_{ab}=b/a$, $0 < r_{ab} \le 1$ forms the ellipse inside the circle; i.e., the axes of the ellipse produced are major axis $a = R$ and minor axis $b = r_{ab}*R$. From Fig. 2 we can relate the new maximum AOA of the ellipse with the maximum AOA of the circle case by the following expression:

$$r_{ab}\tan(\varphi_{c\max})=\tan(\varphi_{e\max}), \qquad (5)$$

Using (2) and (3) we can obtain a general expression of the maximum AOA for both cases described as follows:

$$\varphi_{e\max}=\arctan\left(r_{ab}\tan\left(\arcsin\left(\frac{a}{R_c}\right)\right)\right), \qquad (6)$$

We can note from (4) that it becomes identical to (2) for the particular case of $r_{ab} = 1$ (Circular case). We also note that the area *ABCD* (shaded area within the ellipse in Fig. 2) defined by the x-axis and the line described by points *ODC* is in function of the angleφ, within the range $0 < \varphi \le \varphi_{emax}$, i.e., *A(φ)*. Then for a uniform distribution of the scatterers inside the cluster the cumulative density function (*CDF*) of the AOA can be defined as follows:

$$F_{\varphi}(\varphi)=\frac{A(\varphi)}{\frac{1}{2}A_{e}}=\frac{2A(\varphi)}{ab\pi}, \tag{7}$$

Where A_e = $ab\pi$ denotes the area of the ellipse. The equation of the ellipse defined in Fig. 2 is

$$\frac{\left(x-R_{c}\right)^{2}}{a^{2}}+\frac{y^{2}}{b^{2}}=1, \tag{8}$$

Which we can transform (6) into polar coordinates using and rearranging the relations defined in (1), obtaining a second order equation:

$$r^{2}\left(\frac{\cos^{2}(\varphi)}{a^{2}}+\frac{\sin^{2}(\varphi)}{b^{2}}\right)-r\left(\frac{2R_{c}\cos(\varphi)}{a^{2}}\right)+\frac{R_{c}^{2}}{a^{2}}-1=0\,, \tag{9}$$

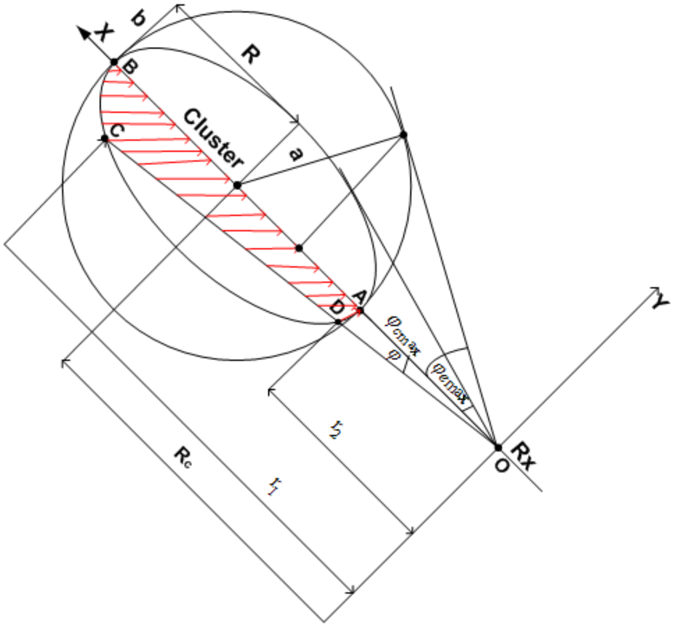


Fig. 2. Geometry of the model for calculating the PDF of the AOA.

Solving (7) with respect to *"r"*, we obtain the following expressions, corresponding to the two radii r_1 and r_2 shown in Fig. 2:

$$r_1=$$
$$\frac{R_{c}b^{2}\cos(\varphi)+\sqrt{b^{4}a^{2}\cos^{2}(\varphi)-b^{2}a^{2}\sin^{2}(\varphi)R_{c}^{2}+b^{2}a^{4}\sin^{2}(\varphi)}}{b^{2}\cos^{2}(\varphi)+a^{2}\sin^{2}(\varphi)}, \tag{10}$$

and

$$r_2=$$
$$\frac{R_{c}b^{2}\cos(\varphi)-\sqrt{b^{4}a^{2}\cos^{2}(\varphi)-b^{2}a^{2}\sin^{2}(\varphi)R_{c}^{2}+b^{2}a^{4}\sin^{2}(\varphi)}}{b^{2}\cos^{2}(\varphi)+a^{2}\sin^{2}(\varphi)}, \tag{11}$$

An area bounded in function of $r_1(\varphi)$ and $r_2(\varphi)$ in polar coordinates is then given by the following expression:

$$A(\varphi)=\tfrac{1}{2}\int_{0}^{\varphi}\left(r_{1}^{2}(\xi)-r_{2}^{2}(\xi)\right)d\xi, \tag{12}$$

Next, inserting (8) and (9) into (10) and the result into (5), we obtain the CDF of the AOA defined by the following integral:

$$F_{\varphi}(\varphi)=$$
$$\int_{0}^{\varphi}\frac{4bR_{c}\cos(\xi)\sqrt{-a^{2}b^{2}(-\cos^{2}(\xi)b^{2}+R_{c}^{2}-R_{c}^{2}\cos^{2}(\xi)-a^{2}+a^{2}\cos^{2}(\xi))}}{a\pi(\cos^{2}(\xi)b^{2}+\sin^{2}(\varphi)a^{2})^{2}}, \tag{13}$$

Finally, the PDF of the AOA can be calculated by taking the derivative of the CDF with respect to φ, obtaining the following expression:

$$f_{\varphi}(\varphi)=\frac{d}{d\varphi}F_{\varphi}(\varphi)=$$
$$\begin{cases}\dfrac{4r_{ab}R_{c}\cos(\varphi)\sqrt{-a^{2}b^{2}(-\cos^{2}(\varphi)b^{2}+R_{c}^{2}-R_{c}^{2}\cos^{2}(\varphi)-a^{2}+a^{2}\cos^{2}(\varphi))}}{\pi(\cos^{2}(\varphi)b^{2}+\sin^{2}(\varphi)a^{2})^{2}}, \\ for-\varphi_{\max}<0\leq\varphi_{\max} \\ 0,\;\; elsewhere \end{cases} \tag{14}$$

Since the clusters are bounded by a circle when r_{ab} =*1*; i.e., for the case *b*=*a*, the AOA of the MPCs (those that make up the same cluster) at the Rx is restricted to an angular region of $2\varphi_{cmax}$, as illustrated in Fig. 2.

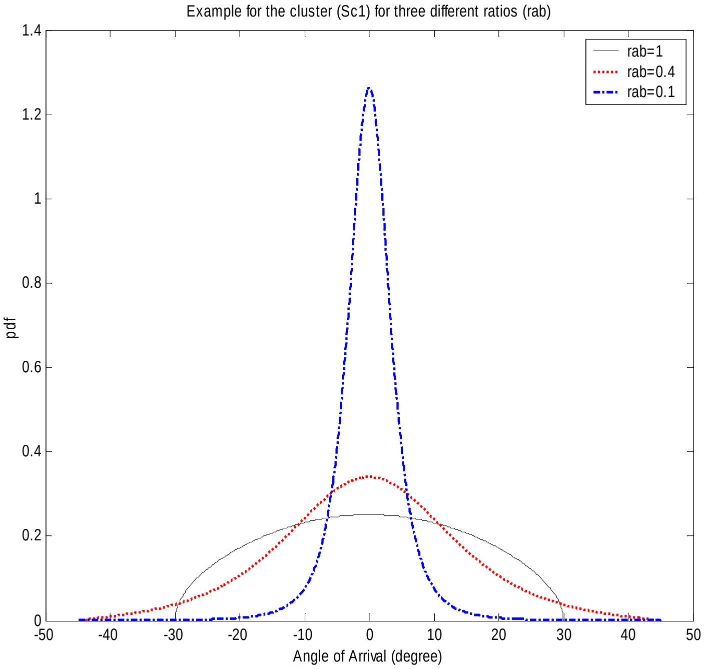
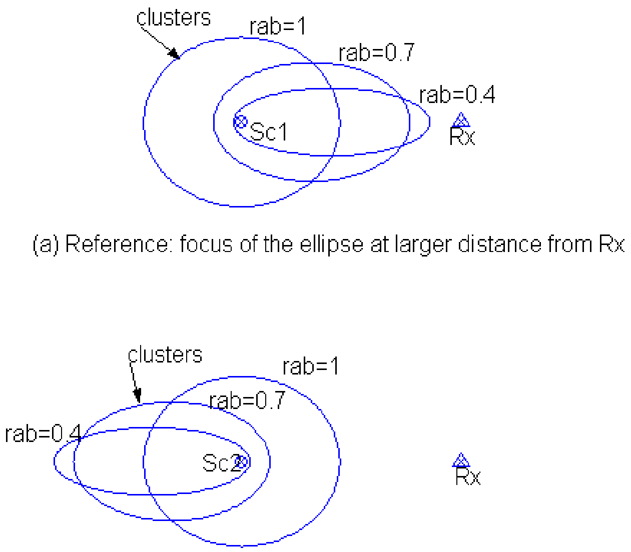


Fig. 3. PDF of AOA for a cluster with three different ratios: r_{ab}=1, r_{ab} = 0.4, and r_{ab}=0.1 bounded by a circular cluster, using as a reference the centre of the ellipse as shown in fig.1.



(a) Reference: focus of the ellipse at larger distance from Rx

(b) Reference: focus of the ellipse at closer distance from Rx

Fig. 4. Area of a cluster with three different ratios: $r_{ab}=1$, $r_{ab} = 0.7$, and $r_{ab}=0.4$. Reference: the separation distances between the cluster (Sc) and the Rx, the foci of the ellipses (a) *Sc1* and (b) *Sc2*, respectively.

Fig. 3 shows one example for the PDF of the AOA, for two different shapes of clusters: for the case of a circular cluster ($r_{ab}=1$), and for two elliptical clusters for the cases $r_{ab}=0.4$ and $r_{ab}=0.1$, respectively. We note from Fig. 2 that the AOA is the maximum for the circular case, as expected from (4) and Fig. 2. For the elliptical cases, we note that the AOA decreases as the ratio r_{ab} decreases. This is always valid when the circular cluster bounds the ellipses, i.e. when $r_{ab}=1$, and the separation distance between the centre of the cluster and the Rx is fixed, as illustrated in Fig. 2.

On the other hand, in our application we consider two cases based on the foci of the ellipses as illustrated in Fig. 4. One case is when we use as a reference for the distance between the Cluster and the Rx the focus of the ellipse (Sc1) situated at larger distance from the Rx as shown in Fig. 4(a). The other case is when we use as a reference the focus of the ellipse (Sc2) situated at closer distance from the Rx as shown in Fig. 4(b).

IV. Comparison with Experimental Results

A. Simulation Results

Let us now validate the theoretical pdf using some numerical examples. The theoretical PDF for the AOA described in (12) is evaluated for a test case where the separation distance between the base station as Tx and the mobile unit as Rx is 600 meters, and the assumed separation distance between the cluster and the Rx is 224 meters, which are typical values of distances for outdoors scenarios in urban environments such as City-street scenario or Highway–scenario, see [4], and [11], where the scenarios differ mainly in the size of the environment and the cluster density. The (x, y) position of two clusters at the same distance from the Rx is to show the two cases analyzed in the previous section as illustrated in Fig. 5, the results of the power delay angle profiles (PDAPs) obtained from the cluster positions generated are presented in Fig. 6, Note that the AOA is decreasing when we use as a reference the focus of the ellipse situated at closer distance from the receiver as shown previously in Fig. 4(b).

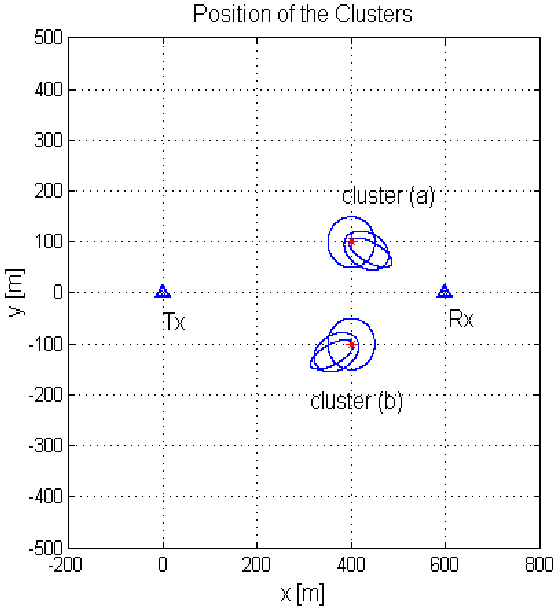


Fig. 5. X-Y Cluster position for three different ratios: $r_{ab}=0.4$, $r_{ab}=0.7$, and $r_{ab}=1$, respectively.

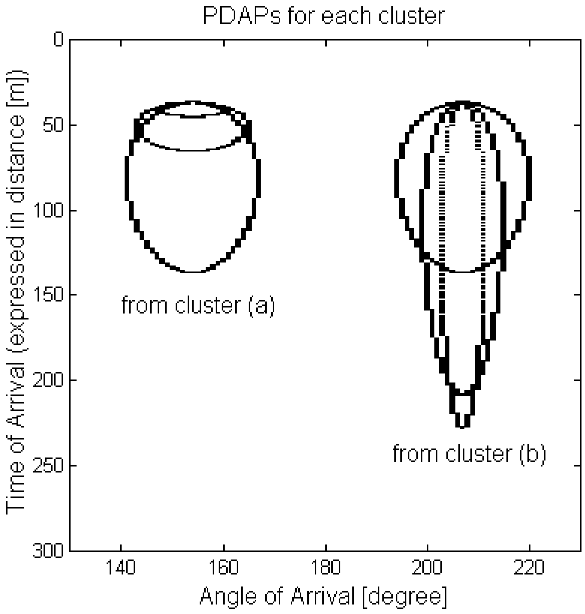


Fig. 6. PDAPs: horizontal axisα, vertical axis delay expressed in meters, and α_{LOS}= 180 deg.

B. Comparison with Published Results

Several experimental results are available to which we can compare our theory. In the indoor case, Chong et al. [6] have characterized the indoor wideband channel model to the angular domain through experimental results obtained by a wideband vector channel sounder together with an eight-element uniform linear array receiver (Rx). MPC parameters were estimated using a super-resolution frequency domain algorithm (FD-SAGE) and clusters were identified in the spatial-temporal domain by a nonparametric density estimation procedure. The clustering effect also gives rise to two classes of channel power density spectra (PDS)-intercluster and intracluster PDS, which are shown to exhibit Laplacian function in the angular domain, such as the power angular spectrums (PASs).

TABLE I
EXPERIMENTAL RESULTS FROM [4] IN ANGLE (DEGREE.) AND DELAY (EXPRESSED IN DISTANCE (M)) OF THE PDAPS FOR EACH CLUSTER

Cluster No.	Excess Delay [m]	Delay Spread [m]	Angle-of-arrival α (degrees)	Angle Spread 2Δα (degrees)
Sc1	150	60	-8	25
Sc2	45	60	-6	6
Sc3	15	60	0	8
Sc4	210	180	0	8

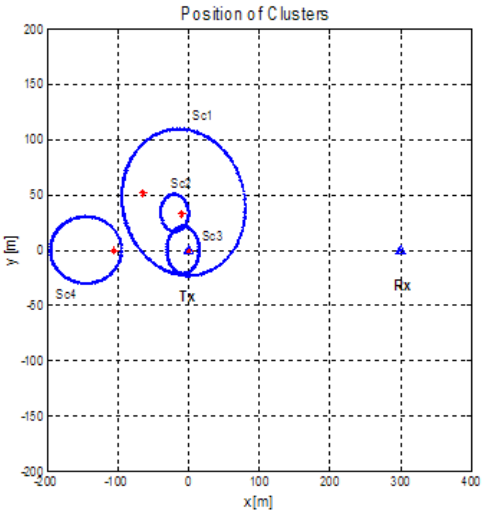


Fig. 7. X-Y Cluster's position obtained, using the experimental results PDAPs from [4] and the double bounce approach as described in [7].

In the outdoor case, Toeltsch et al. [4] used a wideband channel sounder together with a planar antenna array.determine the parameters of the incident waves. A super-resolution algorithm (Unitary ESPRIT) allows resolving individual MPCs in such clusters and hence enables a detailed statistical analysis of the propagation properties. We make comparison with one of the experimental results published in [4], summarized in TABLE I. In this TABLE we present the clusters parameters extracted from the measurement campaign, i.e., the excess delay, delay spread, (both in terms of distance), AOA, and angle spread, (both in degrees). Then, in Fig. 7 we plotted the position of each cluster based on the measurements of the PDAPs published in [4]. In Fig. 8 we show the boundaries of PDAPs for each cluster obtained from the set of parameters extracted from [4] and defined in TABLE I. As shown in Fig. 7, we can describe different shapes and sizes of clusters found in the PDAPs from measurement campaigns published in the open literature, as in [4].

V. Conclusions

This paper presented analytical expressions for the AOA power density function (PDF) and its application in geometrically based channel models using the clustering approach model described in [7].

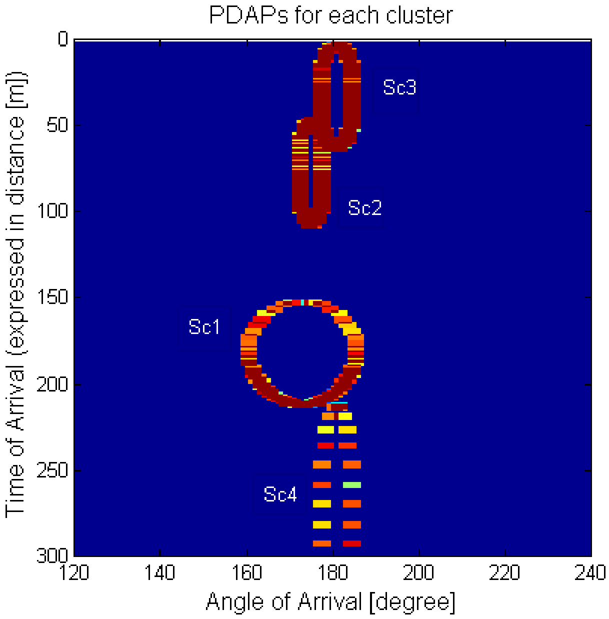


Fig. 8. Boundaries of the PDAPs for each cluster: horizontal axisα, vertical axis delay expressed in meters, and αLOS= 180 deg.

The average number of clusters and the MPCs distribution within a cluster depend heavily on the resolution of the parameter estimation algorithm. They also depend on the type of scenario, (indoor or outdoor, i.e., the size of the environment and the cluster density). Furthermore, as stated in [6], the number of clusters and MPCs detected also depend on several others factors such as the Tx-Rx separation and location, the physical layout of the environment and the dynamic range of the channel sounder.

Acknowledgements

This work is supported by Swedish International Development Cooperation Agency (SIDCA).

REFERENCES

[1] X. Zhao, J. Kivinen, P. Vainikainen, and K. Skog, “Propagation characteristics for wideband outdoor mobile communications at 5.3 GHz,” *IEEE J. Select. Areas Commun.*, vol. 20, No. 3, pp. 507–514, Apr. 2002.

[2] K. I. Pedersen, P. E. Mogensen, and B. H. Fleury, “A stochastic model of the temporal and azimuthal dispersion seen at the base station in outdoor propagation environments,” *IEEE Trans. Veh. Technol.*, vol. 49, pp. 437-447, Mar. 2000.

[3] J. Fuhl, A. F. Molisch, and E. Boneck, “Unified channel model for mobile radio systems with smart antennas,” in *Proc. Inst. Elect. Eng.-Radar Sonar Navigation*, vol. 145, Feb. 1998, pp. 32-41.

[4] M. Toeltsch, J. Laurilla, K. Kalliola, A. F. Molisch, P. Vainikainen, and E. Bonek, “Statistical characterization of urban spatial radio channels,” *IEEE Journal on selected areas in Communications*, vol. 20, No. 3, pp 539-549, April 2002.

[5] J. P. Kermoal, L. Schumaker, K. I. Pedersen, P. E. Mogensen, and F. Frederiksen, “A stochastic MIMO radio model with experimental validation,” *IEEE Journal on selected areas in Communications*, vol. 20, No. 6, pp 1211-1226, June 2002.

[6] C. C. Chong, Ch. M. Tan, D. I. Laurenson, and S. McLaughlin, “A New Statistical Wideband Spatio-Temporal Channel Model for 5-GHz Band WLAN Systems,” *IEEE Journal on selected areas in Communications*, vol. 21, No. 2, pp 139-150, February 2003.

[7] M. R. Arias and B. Mandersson, “Clustering Approach for Geometrically Based Channel Models in Urban Environments,” *IEEE Antennas and Wireless Propagation Letters*, vol.5, pp 290-293, December, 2006.

[8] A. F. Molisch, “A Generic Model for MIMO Wireless Propagation Channels in Macro-and Microcells,” *IEEE Transactions on Signal Processing*, vol. 52 No. 1, pp 61-71, January 2004.

[9] J. B. Andersen and K.I. Pedersen, “Angle-of-Arrival Statistics for Low Resolution Antennas,” *IEEE Transactions on Antennas and Propagation,* vol. 50, No. 3, March 2002.

[10] M. R. Arias and B. Mandersson, “A Generalized Angle Domain Clusters PDF and Its Application in Geometrically Based Channel Models,” in *Proc. Fifth International Conference on Information, Communications & Signal Processing, ICICS 2005*, vol.1, December, 2005, pp 1339-1343.

[11] J. M. Gil, F. C. Cardoso, B. W. Kuipers, and L. M. Correia, “Contribution for the Definition of Common Propagation Scenarios”, *IST-NEWCOM Project Repo*rt, Lisbon, Portugal, May 2005.

Journal Information and Instructions for Authors

I. Journal Information

"*Polibits*" is a half-yearly research journal published since 1989 by the Center for Technological Design and Development in Computer Science (CIDETEC) of the National Polytechnic (Technical) Institute (IPN) in Mexico City, Mexico. The journal solicits original research papers in all areas of computer science and computer engineering, with emphasis on applied research.

The journal has double-blind review procedure. It publishes papers in English and Spanish.

Publication has no cost for the authors.

A. Main Topics of Interest

The journal publishes research papers in all areas of computer science and computer engineering, with emphasis on applied research.

More specifically, the main topics of interest include, though are not limited to, the following:

– Artificial Intelligence
– Natural Language Processing
– Fuzzy Logic
– Computer Vision
– Multiagent Systems
– Bioinformatics
– Neural Networks
– Evolutionary algorithms
– Knowledge Representation
– Expert Systems
– Intelligent Interfaces: Multimedia, Virtual Reality
– Machine Learning
– Pattern Recognition
– Intelligent Tutoring Systems
– Semantic Web
– Database Systems
– Data Mining
– Software Engineering
– Web Design
– Compilers
– Formal Languages
– Operating Systems
– Distributed Systems
– Parallelism
– Real Time Systems
– Algorithm Theory
– Scientific Computing
– High-Performance Computing
– Geo-processing
– Networks and Connectivity
– Cryptography
– Informatics Security
– Digital Systems Design
– Digital Signal Processing
– Control Systems
– Robotics
– Virtual Instrumentation
– Computer Architecture
– other.

B. Indexing

LatIndex, Periódica, e-revistas.

II. Instructions for Authors

A. Submission

Papers ready to review are received through the Web submission system www.easychair.org/polibits. See also the updated information at the web page of the journal www.cidetec.ipn.mx/polibits.

The papers can be written in English or Spanish.

Since the review procedure is double-blind, the full text of the papers should be submitted without names and affiliations of the authors and without any other data that reveals the authors' identity.

For review, a file in one of the following formats is to be submitted: PDF (preferred), PS, Word. In case of acceptance, you will need to upload your source file in Word or TeX. We will send you further instructions on uploading your camera-ready source files upon acceptance notification.

Deadline for the nearest issue (January-June 2011): April 1, 2011. Papers received after this date will be considered for the next issues.

B. Format

Please, use IEEE format[1], see section "Template for all Transactions (except IEEE Transactions on Magnetics)". The editors keep the right to modify the format and style of the final version of the paper if necessary.

We do not have any specific page limit: we welcome both short and long papers, provided the quality and novelty of the paper adequately justifies the length.

In case of being written in Spanish, the paper should also contain the title, abstract, and keywords in English.

[1] www.ieee.org/web/publications/authors/transjnl/index.html