Smart Acess Control Model

Akash K Singh, PhD
IBM Corporation Sacramento, USA

Abstract

For century, there has been no change in the fundamental structure of the electrical power grid and vehicle networks. Current hierarchical, centrally controlled grid of the electrical grid is not best for growing demand. To address the challenges of the existing power grid, the new concept of smart grid and smarter planet are under research. The smart grid can be considered as a modern electric power grid infrastructure for enhanced efficiency and reliability through automated control, high-power converters, modern communications infrastructure, sensing and metering technologies, and modern energy management techniques based on the optimization of ondemand, energy and network availability. While current power systems are based on a solid information and communication infrastructure, the new smart grid needs a different and much more complex one, as its dimension is much larger and needs utmost performance. This paper addresses critical issues on smart grid technologies primarily in terms of information and communication technology (ICT) issues and opportunities. The main objective of this paper is to provide a contemporary look at the current state of the art in smart grid communications as well as to discuss the still-open research issues in this field. It is expected that this paper will provide a better understanding of the technologies, potential advantages and research challenges of the smart grid and provoke interest among the research community to further explore this promising research area.

Keywords- Advanced metering infrastructure (AMI), communication technologies, quality-of-service (QoS), smart grid, standards

I. INTRODUCTION

World is moving toward Smart Grid which is an area of research and future that can change the energy and utility systems, we will see the paradigm shift by 2030, Electrical and Power Systems are getting networked with Sensors, appliances and devices at home, office and city and transportation vehicles. The Smart Grid is a network of network and an initiative to revolutionize the Telecommunication and Electric sector and Network devices. Sensors, appliances and devices are deployed around the world and connected on IP Network; more over it's an intelligent network. Smart grid network predicts and intelligently manage the electric power, resource usage and connects the power and energy and utility producer and consumer to achieve the high reliability, quality of services, economy at large and electrical/ power services at low cost and reduction in wastage of energy. Market is facing that there is a potential growth in electricity market to provide value added services, on demand energy resources with lower prices to the community and industrial sector. The Smart grid addresses the challenges of developing efficient utility management and services, assets management, industrial and home automation, and wireless communication technologies to integrate wide variety protocols and device APIs and on-Demand services. The distributed grids will support bidirectional power flows, active power balancing and ancillary services like stability in voltage will be in demand with overall object to deliver energy to end users, with this grid voltage and current remain as constant.

The Smart Grid framework transforms the US electric power system into more interoperable Smart grid where communication systems and network systems get integrated with power delivery infrastructure; it enables the two way flows of energy and communications to connected devices.
Devices and applications functions as gateway to interface energy providers and consumer and are distributed geographically across consumer base and networked locations. It is required to have secured communication between the consumer devices and the energy providers, systems architecture supports the interchangeability between standards and associated interfaces and integration points, Interoperability of multiple networks, systems, devices, and applications communicate and exchange information in secured link and able to transform the message formats and adhere communication standard protocol. Smart grid framework supports the interoperable systems with agreed response format from other components of grid system, and able to perform under established service level agreement and quality of service and high availability and scalability. Cyber security and network communication are key element of smart grid framework and there is a vast research underway to develop new security algorithms and techniques that can create larger encryption keys and keep the better performance.

Wide variety of multiple networks are being established such as Industrial Area Network, Building Area Network and Home Area Network , Energy management system and infrastructure for Industry, commercial, building, city and home, geographical area network for Metering, Infrastructure and Field sensors and devices that can monitor and perform analytics. Most pressing challenges are to build an Intelligent Network that can manage large number of devices and able to have efficient communication networks interms of routing, switching and capacity management of traffic and manage nodes. The Energy grid improvement requires the universal adaption of Standards and Framework that would integrate the legacy devices, applications and energy girds with future network and sensor devices. Broadband communication network and wireless communication needs to improve to support distributed intelligence in Electrical and Energy sectors and be able to handle high speed communication channel. As these systems are integrated and could lead the security risk, so its important to develop the security architecture for Smart grid as well that can provide higher level of security with optimal performance. Internet protocol (IPv4 and IPv6) are baseline protocol for the network communication and IPsec and Transport layer security (TLS) adds security layer on IP Network, Networks do provide the multi homing principles and High Performance Infrastructure.

A. PKI Data Communication Protocol for Building Automation and Control Networks

The security requirements for smart grid, as well as the scale of the system and availability required at all the time, Usage of public key infrastructure (PKI) technologies along with trusted computing elements, supported by other architectural components, is the best overall solution for smart grid. The most effective key management solution for securing the smart grid will be based on PKI technologies and PKI Infrastructure. PKI is more than just the hardware and software in the system, there are appliances that provide the key management and PKI. This also includes the policies and procedures which describe the set up, management, updating, and revocation of the certificates that are at the heart of PKI [4]. A PKI binds public keys with user identities through use of digital certificates where x.509 certificate is installed on the appliance. The binding is established through a registration process, where after a registration authority (RA) assures the correctness of the binding, the certificate authority (CA) issues the certificate to the user. Users or devices can authenticate each other via the digital certificates, establish symmetric session keys, and subsequently encrypt and decrypt messages between each other with the basic steps in utilizing a PKI. The certificate subject, desiring communication with a secure resource begins by sending a certificate signing request (CSR) of 1024 or 2048 bit to the RA. The RA performs a vetting function which determines if the requested bindings are correct, and if so signs the CSR and forwards it to the CA, which then issues the certificate. Later when the certificate subject wishes to access a secure resource, it sends the certificate to the RP. The RP validates the certificate typically by requesting the certificate status from a validation authority (VA), who replies in the positive if the certificate is valid. PKI allows for a chain of trust, where a first CAs extends trust to a second CAs by simply issuing a CA-certificate to the second CAs. This enables RPs that trusts the first CA to also trust subjects with certificates issued by the second CA. When two CAs issue each other certificates it is referred to as cross signing. In this way, CAs from one organization can extend trust to the CAs from other organizations, thus enabling secure interoperability across domains. CA certificates can contain various constraints to limit the trust being extended by the issuing CA to the subject CA. In very large systems PKI could be significantly more efficient than shared keys in terms of setting up and maintaining operational credential. This is due to the fact that each entity needs to be configured with its own certificate. This is as compared to symmetric key provisioning where each device may need to be configured with a unique key pair for every secure link. While PKI is known for being complex, many of the items responsible for the complexity can be significantly reduced by including the following four main technical elements:

- PKI standards
- automated trust anchor security;
- certificate attributes;
- smart grid PKI tools.

Standards are used to establish requirements on the security operations of energy service providers (e.g., utilities, generators).

B. Network Availability

The Malicious attacks targeting network availability can be considered as denial-of-service (DoS) attacks, which attempt to delay, block or corrupt information transmission in order to make network resources unavailable to nodes that need information exchange in the smart grid. Since it is widely expected that at least part, if not all, of the smart grid will use IP-based protocols (e.g., IEC 61580 [9] has already adopted TCP/IP as a part of its protocol stacks) and TCP/IP is vulnerable to DoS attacks, sophisticated and efficient countermeasures to DoS attacks are essential to the smart grid. DoS attacks against TCP/IP have been well studied in the literature regarding attacking types, prevention and response [12]–[14]. Therefore, in the following, we will discuss potential attacks that specifically target power network availability. As aforementioned, a major difference between the smart grid and the Internet is that the smart grid is more concerned with the message delay than the data throughput due to the timing constraint of messages transmitted over the power networks. Indeed, network traffic in power networks is in general time-critical. For instance, the delay constraint of generic object oriented substation events (GOOSE) messages is 4 ms in IEC 61850 [3]. Such a timing constraint ensures reliable monitoring and control of power devices. But on the other hand, it becomes one of the most vulnerable parts in power networks to DoS attacks. More specifically, instead of using some extreme means (e.g., channel jamming), an attacker can even use legitimate methods to intentionally delay the transmissions of time-critical messages to violate the timing requirements. For instance, an attacker can physically connect to a communication channel in a power network and generates legitimate but useless traffic to capture the channel and to delay the transmission of power monitoring and control devices Since intruders only need to connect to communication channels rather than authenticated networks in the smart grid, it is very easy for them to launch DoS attacks against the smart grid, especially for the wireless-based power networks that are susceptible to jamming attacks [15]–[17]. Hence, it is of critical importance to evaluate the impact of DoS attacks on the smart grid and to design effective countermeasures to such attacks. We will provide initial experimental results of the impact of DoS attacks on the performance of a power network.

II. CONVERSATION BASED ACCESS CONTROL

In this section, we introduce the basic concepts of Web service conversations, credentials, and access control. We then elaborate on the fundamental concepts underlying the proposed approach.

A. Conversational Model for Webservices

We represent the behavioral semantics of a Web service as the set of operations it exports and constraints on the possible conversations clients can execute. For a large class of Web services, as discussed in Benatallah et al. [2003], all such aspects can be represented as a finite transition system (refer to the WSMO choreography concept). We do not provide details about the semantic description of an operation since it is outside the scope of our work. Such semantics would be typically expressed according to a given OWL/OWL-S/WSMO ontology. Thus, a service is generally characterized by two facets: (i) static semantics, dealing with messages, operations, and parameters exchanged and their types, and (ii) dynamic semantics, dealing with the correct sequencing of operations that represents the external workflow offered by the service. The focus of our work is on the dynamic semantics.

B. Access Control Model

Credentials are the mean to establish trust between a client and a service provider. Credentials contain assertions about properties qualifying a given client, referred to as the owner. They are issued by a trusted Certification Authority (CA), which has the required domain expertise to assert that the credential owner has the set of attributes listed in the credential. The CA signs the credential with its private key so that when the credential owner uses the credential for authentication purposes, the service provider verifies the signature of the CA on the credential by using the CA's public key.

Definition 2.2 (Credential). A credential C is a tuple (Issuer,Owner, T ype,Attr, Sign) where Issuer is the identifier of the CA that issues the credential, Owner is the identifier of the credential owner, T ype denotes the type of the credential, Attr = {A1, . . . ,An} is the set of attributes characterizing the credential type T ype, and Sign is the signature of the Issuer on the whole credential. An attribute Ai is a pair (nameAi, valueAi), where nameAi is the name of the attribute Ai and valueAi is a value in the attribute domain domAi of Ai.3 The invocation of the operations provided by a Web service is protected by access control policies defined by the Web service provider. These policies define conditions that clients' credentials would have to satisfy for the client to be granted the right to execute a given operation. Operation access control policies are formally defined as follows.

C. Conversational and Trust Policies

In our approach, clients interact with a Web service by invoking operations according to a conversation. Clients are usually interested in conversations that lead to some final states. We refer to these conversations as meaningful conversations.

Definition 2.7 (Meaningful Conversations). Let WS = (_, S, s0, δ, F) be a Web service and s be a state in S. The set of meaningful conversations originating from s, denoted asMs, is the set {conv | s conv =⇒ t, t ∈ F}.

To perform meaningful conversations, clients have to provide the credentials specified by the conversation access control policies. Access control policies can contain sensitive information and should be protected from inappropriate access. Sensitive access control policies should not be disclosed until the service provider has established sufficient trust with the clients [Seamons et al. 2001; Yu et al. 2003]. Therefore, to protect conversations' access control policies, we introduce a second type of policies, called trust policies. Trust policies specify conditions on the credentials submitted by the client that must be satisfied to entrust the client with a set of conversation access control policies. Since conversation access control policies can have different sensitivities, the Web service provider groups them into sets based on these sensitivities. A trust policy is defined for each set of conversation access control policies.

Definition

The satisfaction of a trust policy by a client will hence define the level of trust that the Web service has on the client in state s. Based on the notion of trust policy, we introduce the notion of allowable conversations, that is, the set of meaningful conversations that are protected by the conversation access control policies which in turn are protected by the trust policy satisfied by the client. Furthermore, we introduce the concept of granted conversations, that are the allowable conversations that start with the execution of op, that is, the specific operation the client has requested.

To summarize, in our approach we consider three different sets of conversations.

—The set of meaningful conversations associated with a state s of WS is static and computed offline (not at enforcement time). Its computation is necessary for the Web service provider to determine the conversations for which it has to define access control policies. Once access control policies are defined, the Web service provider groups the policies according to their sensitivity.

—The set of allowable conversations is dynamically associated with a client in a given state of the interaction with the Web service. The Web service provider assigns the client a set of conversation access control policies, the trust policies of these are satisfied by the client's credentials. The meaningful conversations protected by the conversations's access control policies the client is entrusted with are the allowable conversations.

—The set of granted conversations is dynamically associated with a client in a given state of the interaction with theWeb service. The set of granted conversations is the subset of allowable conversations starting with the operation chosen by the client, and for which the client's credentials satisfy the corresponding conversation access control policies.

D. Access Control Enforcement

In this section, we describe two important aspects of ACConv: computation of meaningful conversations and ACConv's access control enforcement protocol.

E. Conversations Computation

To enforce the proposed access control model, aWeb service provider has to compute for each state s of the Web service all the possible meaningful conversationsMs. Once, for a state s, the meaningful conversations Ms are determined, the Web service provider specifies the access control policies for each operation and then derives the conversation access control policy for the meaningful conversations Ms. The conversation access control policies can have different sensitivity according to the level of protection required by the conversations. Therefore, the Web service provider groups the conversation access control policies associated with state s in sets Cls of policies having the same sensitivity level. The disclosure of each set of access control policies Cls is protected by a trust policy Ps Cls that is defined by the Web service provider. The number of meaningful conversations for a state s is finite if the Web service transition system is acyclic. The meaningful conversations can be computed by a simple breadth-first traversal of the transition system. However, if the transition system contains cycles, the number of meaningful conversations can be potentially infinite and so is the number of conversation access control policies. Fortunately, if a client performs a conversation that involves a cycle, the service provider has to verify that the client is authorized to invoke the operations in the cycle only one time. Since we assume that once client's credentials have been checked, they are valid for the whole conversation execution they have been requested for. Thus, the meaningful conversations in which a cycle is traversed only once are equivalent to the conversations in which a cycle is traversed an infinite number of times. Based on the preceding assumption, we propose an algorithm to compute all the possible meaningful conversations for each state of a Web service transition system. The algorithm is based on the concept of strongly connected component (SCC for short). A strongly connected component is the maximal subgraph of a directed graph such that for every pair of vertexes (u, v) in the subgraph, there is a directed path from u to v and a directed path from v to u [Tarjan 1972]. The transition system of a Web service can be considered as a directed graph where a transition between two states is a directed edge without the label. Therefore, a new acyclic graph can be generated whose nodes represent the different strongly connected components of the initial Web service transition system WS. In the new graph, the cycles are "collapsed" into strongly connected components while the states which are not involved in cycles will remain unchanged in the new graph. In what follows, we denote as c(s), the strongly connected component to which a state s belongs to. The graph of the strongly connected components can be formally defined as follows.

F. Enforcement Protocol

Our access control enforcement protocol ensures that Web service providers disclose to clients only the access control policies of conversations these clients want to perform. Therefore, clients have to provide only the credentials necessary to be granted the execution of the conversations they are interested in. In fact, the Web service providers ask clients to provide the credentials specified in the access control policies of allowable conversations starting with the operation chosen by the clients. If clients provide the requested credentials, they can perform any of the allowable conversations starting with the chosen operation. Since clients are requested to provide in advance all the credentials in order to be authorized to perform conversations starting with the operation they have selected, the risk that clients are not able to progress the interaction with the Web service is minimized. The risk is minimized but not eliminated: clients might still not be able to progress if they want to perform an operation that is not part of the granted conversations. In this case, our enforcement protocol tries to entrust clients with another set of access control policies and with another set of granted conversations. The enforcement process is represented in Figure 1. The enforcement is triggered whenever a Web service receives an invocation of an operation Op from a client. If Op is included in the set of granted conversations, Op is executed and the result is returned to the client. Otherwise, the credentials presented by the client are evaluated against the trust policies of the set of access control policies protecting conversations starting with Op. If the client's credentials satisfy a trust policy, the client is associated with a set of access control policies and with the corresponding set of allowable conversations. Thus, the client is requested to provide the credentials specified in the access control policies of the subset of allowable conversations starting with Op. If the client provides the requested credentials, the client is granted the execution of the allowable conversations starting with Op, that are the granted conversations. If the client does not match any trust policy, the client is entrusted only with the access control policy of operation Op. Therefore, the client is asked to provide only the credentials required by Op access control policy. When the client requests the next operation, the enforcement system tries to entrust the client with a set of access control policies and of granted conversations on the basis of the current operation requested. The Enforcement() algorithm represents the overall enforcement process. The first step is to check if the current operation Op is contained in the set of operations composing the set Granted Conversations. If this is the case, the operation is executed. Otherwise, TrustAssignment() is executed to check if the client can be entrusted with a set of conversation access control policies and of allowable conversations. If this is not the case, the client is entrusted only with PolOp, the operation access control policy associated with Op. Then, the method Select Cred returns the set of credentials listed in the policy PolOp that the client has not yet provided. Thus, the client is requested to submit only the credentials in the set returned by Select Cred. If the client's credentials in set Cred Set satisfy Op policy, Op is performed; otherwise the client is denied the execution of Op. If the client is entrusted with a set of conversation access control policies Pol Set and a set of allowable conversations Conv Set, the algorithm first computes the subset Conv of Conv Set containing the conversations that start with the execution of Op and the subset Pol of Pol Set containing the access control policies of conversations in Conv. Then, the client is asked to provide the credentials listed in the access control policies in Pol. The set Conv of conversations which access control policies are satisfied by the client's credentials becomes the new set of granted conversations. If the client does not satisfy any of the policies in Pol, the algorithm checks if client's credentials satisfy the access control policy associated with Op. If this is the case, the execution of Op is granted to the client, otherwise the interaction with the client is terminated. The algorithm isGranted() checks if the operation Op invoked by the client is contained in the set of operations composing the conversations in set Granted Conversations. The TrustAssignment() algorithm determines which trust policy PsCls I is satisfied by Client Cred, that is, the set of credentials submitted by the client prior to state s. If the client's credentials satisfy a trust policy Ps Cls i, the client is entrusted with the set of access control policies Cls i and with the set of allowable conversations As Clsi . Otherwise, the client is entrusted only with the access control policy POp associated with the operation Op chosen by the client and with an empty set of allowable conversations. We refer the reader to Appendix A for a complete example of enforcement based on operation and conversation access control policies defined for the Amazon Flexible Payment Web service.5

III. ENFORCEMENT SYSTEMS

The system architecture of ACConv is compliant with the XACML standard [Moses 2005]. The main components of the access control enforcement system are a Policy Enforcement Point (PEP), a Policy Decision Point (PDP), and a Policy Administration Point (PAP). With respect to the XACML architecture, we have added a component called Execution Controller System (ECS), see Figure 2. The ECS provides the security administrator with a simple graphical interface, through which she can compute the meaningful conversations and define trust and access control policies. In particular, the ECS interface allows the security administrator to specify the access control policies for the operations provided by the Web service. Then, the ECS combines operation access control policies to obtain the conversation access control policies. Once the conversation access control policies are computed, the security administrator can group the policies based on their sensitivity and can define a trust policy to protect the release of each set of policies. Access control policies and trust policies are then stored in two repositories that are managed by the PAP component. Besides providing functions to set up the enforcement access control process, the ECS also tracks, at runtime, the state of the interaction between the client and the service. The PEP Module is the interface between the Web service's clients and the ECS. According to the enforcement process described in Section 3, when the client invokes the first operation, it sends a request message along with a set of initial credentials. Once the PEP receives the message, it stores the credentials in a local repository and forwards to the ECS the name of the operation selected by the client. The ECS updates the state of the interaction and returns to the PEP the identifiers of the possible sets of access control policies the client can be entrusted with in that state and the corresponding sets of allowable conversations (steps 2–3). Then, the PEP sends to the PDP the identifier of the sets of conversation access control policies and the sets of allowable conversations (step 4). The PDP's Trust Level Assignment (TLA) module interacts with the PAP which manages the policies, to retrieve the trust policies associated with the current state. Then, it queries the credential repository to evaluate clients' credentials against trust policies. The client is entrusted with the set of conversation access control policies of which the trust policy is satisfied. The TLA module notifies to the Policy Selection (PS) module the identifier of the set of policies the client has been associated with and the set of allowable conversations. The PS module asks the PAP to retrieve the conversation access control policies in the policies set assigned to the client (steps 8–9). Then, the PS module returns the set of policies to the PEP with the set of allowable conversations (step 10). Thus, the PEP asks the client to provide the credentials required by the policies that it has not yet provided and evaluates them against the policies (steps 11–12). If the client's credentials satisfy the policies, the client can perform any operation that composes the granted conversations. Then, the PEP triggers the execution of the operation by the internal middleware and returns the result to the client (steps 13–15). As the PEP stores the granted conversations, every time the client invokes an operation, it first checks if the operation is one of those that compose the granted conversations. If this is the case, the PEP sends to the ECS the name of the requested operation and then triggers the execution of the operation. Otherwise, the PEP requests the ECS to provide the set of the identifiers of the sets of access control policies, and the sets of allowable conversations the client can be entrusted with in the current state of the interaction.

A. Access Control for Composite Webservices

A key strength of Web services is at they can be composed to build new Web services called composite services. The Web services being composed are usually referred to as component services. Composition involves two different aspects [Berardi et al. 2005]: (1) synthesis which is concerned with producing a specification, called composition schema, of how to coordinate the component services to fulfill the client request; and (2) orchestration which relates to the enactment of the composite service and the coordination among component services by executing the composition schema. The composition schema can be generated manually by a designer or (semi-)automatically [Berardi et al. 2005]. An orchestration engine can then invoke the component services according to the schema. In many cases, the behavior of the composite service and the component services can be modeled using transition systems. A transition system is also used to represent the composition schema [Berardi et al. 2005].

B. Architecture

The system architecture for composite Web services is composed of multiple access control enforcement systems, one for each component service, and of a repository for for the component Web services is similar to the system that we have described in Section 4.1 for single Web services. The orchestration engine has two functions: it manages clients' credentials and invokes the component Web services's operations necessary to fulfill a clients' request. When a client invokes an operation (step 1), the orchestration engine first stores in its local repository the credentials the client sent with the invocation. Then, the orchestration engine contacts the component service enforcement system that according to the composition schema is entitled to perform the operation. The PEP of the component service checks if the client is authorized to execute the operation. If this is the case, the operation is performed; otherwise the PDP returns to the PEP the entrusted conversations' access control policies. The PEP sends to the orchestration engine the request for the credentials specified in the conversation access control policies (step 3). The orchestration engine, then, determines which of the credentials, among those requested by the component Web services, are stored in its local credential repository and thus only requests to the client the credentials that the client has not yet provided (step 4). Once the client provides the credentials (step 5), the orchestration engine sends them to the component service's PEP (step 6). If the client's credentials are compliant with the access control policies, the operation is executed and the result is returned to the client through the orchestration engine (steps 7–8). Otherwise, the access is denied.

We consider the following anycast field equations defined over an open bounded piece of network and /or feature space $\Omega \subset R^d$. They describe the dynamics of the mean anycast of each of p node populations.

$$\begin{cases} (\frac{d}{dt}+l_i)V_i(t,r)=\sum_{j=1}^{p}\int_{\Omega} J_{ij}(r,\bar{r})S[(V_j(t-\tau_{ij}(r,\bar{r}),\bar{r})-h_{|j})]d\bar{r} \\ \qquad +I_i^{ext}(r,t), \qquad t\geq 0, 1\leq i\leq p, \\ V_i(t,r)=\phi_i(t,r) \qquad t\in[-T,0] \end{cases} \tag{1}$$

We give an interpretation of the various parameters and functions that appear in (1), Ω is finite piece of nodes and/or feature space and is represented as an open bounded set of R^d . The vector r and $\bar{r}$ represent points in Ω . The function $S: R \to (0,1)$ is the normalized sigmoid function:

$$S(z)=\frac{1}{1+e^{-z}} \tag{2}$$

It describes the relation between the input rate v_i of population i as a function of the packets potential, for example, $V_i = v_i = S[\sigma_i(V_i - h_i)].$ We note V the $p-$ dimensional vector $(V_1,...,V_p)$.The p function $\phi_i, i=1,...,p,$ represent the initial conditions, see below. We note ϕ the $p-$ dimensional vector $(\phi_1,...,\phi_p)$. The p function $I_i^{ext}, i=1,...,p,$ represent external factors from other network areas. We note I^{ext} the $p-$ dimensional vector $(I_1^{ext},...,I_p^{ext})$. The $p\times p$ matrix of functions $J=\{J_{ij}\}_{i,j=1,...,p}$ represents the connectivity between populations i and j, see below. The p real values $h_i, i=1,...,p,$ determine the threshold of activity for each population, that is, the value of the nodes potential corresponding to 50% of the maximal activity. The p real positive values $\sigma_i, i=1,...,p,$ determine the slopes of the sigmoids at the origin. Finally the p real positive values $l_i, i=1,...,p,$ determine the speed at which each anycast node potential decreases exponentially toward its real value. We also introduce the function $S: R^p \to R^p$, defined by $S(x)=[S(\sigma_1(x_1-h_1)),...,S(\sigma_p-h_p))],$ and the diagonal $p\times p$ matrix $L_0 = diag(l_1,...,l_p)$. Is the intrinsic dynamics of the population given by the linear response of data transfer. $(\frac{d}{dt}+l_i)$ is replaced by $(\frac{d}{dt}+l_i)^2$ to use the alpha function response. We use $(\frac{d}{dt}+l_i)$ for simplicity although our analysis applies to more general intrinsic dynamics. For the sake, of generality, the propagation delays are not assumed to be identical for all populations, hence they are described by a matrix $\tau(r,\bar{r})$ whose element $\tau_{ij}(r,\bar{r})$ is the propagation delay between population j at $\bar{r}$ and population i at r. The reason for this assumption is that it is still unclear from anycast if propagation delays are independent of the populations. We assume for technical reasons that τ is continuous, that is $\tau \in C^0(\overline{\Omega}^2, R_+^{p\times p})$. Moreover packet data indicate that τ is not a symmetric function i.e., $\tau_{ij}(r,\bar{r}) \neq \tau_{ij}(\bar{r},r)$, thus no assumption is made about this symmetry unless otherwise stated. In order to compute the righthand side of (1), we need to know the node potential factor V on interval $[-T,0]$. The value of T is obtained by considering the maximal delay:

$$\tau_m = \max_{i,j(r,\bar{r}\in\overline{\Omega\times\Omega})} \tau_{i,j}(r,\bar{r}) \tag{3}$$

Hence we choose $T=\tau_m$

C. *Mathematical Framework*

A convenient functional setting for the non-delayed packet field equations is to use the space $F = L^2(\Omega, R^p)$ which is a Hilbert space endowed with the usual inner product:

$$\langle V,U\rangle_F = \sum_{i=1}^{p}\int_{\Omega} V_i(r)U_i(r)dr \tag{1}$$

To give a meaning to (1), we defined the history space $C = C^0([-\tau_m,0],F)$ with $\|\phi\| = \sup_{t\in[-\tau_m,0]}\|\phi(t)\|F,$ which is the Banach phase space associated with equation (3). Using the notation $V_t(\theta)=V(t+\theta), \theta\in[-\tau_m,0],$ we write (1) as

$$\begin{cases} V(t) = -L_0V(t) + L_1S(V_t) + I^{ext}(t), \\ \qquad V_0 = \phi \in C, \end{cases} \qquad (2)$$

Where

$$\begin{cases} \qquad L_1 : C \to F, \\ \phi \to \int_\Omega J(.,\bar{r})\phi(\bar{r}, -\tau(.,\bar{r}))d\bar{r} \end{cases}$$

Is the linear continuous operator satisfying $\|L_1\| \le \|J\|_{L^2(\Omega^2,R^{p\times p})}$. Notice that most of the papers on this subject assume Ω infinite, hence requiring $\tau_m = \infty$.

Proposition 1.0 If the following assumptions are satisfied.

1. $J \in L^2(\Omega^2, R^{p\times p})$,
2. The external current $I^{ext} \in C^0(R,F)$,
3. $\tau \in C^0(\overline{\Omega^2}, R_+^{p\times p}), \sup_{\overline{\Omega^2}} \tau \le \tau_m$.

Then for any $\phi \in C$, there exists a unique solution $V \in C^1([0,\infty),F) \cap C^0([-\tau_m,\infty,F)$ to (3)

Notice that this result gives existence on R_+, finite-time explosion is impossible for this delayed differential equation. Nevertheless, a particular solution could grow indefinitely, we now prove that this cannot happen.

D. *Boundedness of Solutions*

A valid model of neural networks should only feature bounded packet node potentials.

Theorem 1.0 All the trajectories are ultimately bounded by the same constant R if $I \equiv \max_{t\in R^+} \|I^{ext}(t)\|_F < \infty$.

Proof :Let us defined $f : R \times C \to R^+$ as

$$f(t,V_t) \overset{def}{=} \left\langle -L_0V_t(0) + L_1S(V_t) + I^{ext}(t), V(t) \right\rangle_F = \frac{1}{2}\frac{d\|V\|_F^2}{dt}$$

We note $l = \min_{i=1,\dots p} l_i$

$$f(t,V_t) \le -l\|V(t)\|_F^2 + (\sqrt{p|\Omega|}\|J\|_F + I)\|V(t)\|_F$$

Thus, if

$$\|V(t)\|_F \ge 2\frac{\sqrt{p|\Omega|}.\|J\|_F + I}{l} \overset{def}{=} R,\ f(t,V_t) \le -\frac{lR^2}{2} \overset{def}{=} -\delta < 0$$

Let us show that the open route of F of center 0 and radius R, B_R, is stable under the dynamics of equation. We know that $V(t)$ is defined for all $t \ge 0s$ and that $f < 0$ on ∂B_R, the boundary of B_R. We consider three cases for the initial condition V_0. If $\|V_0\|_C < R$ and set $T = \sup\{t \mid \forall s \in [0,t], V(s) \in \overline{B_R}\}$. Suppose that $T \in R$, then $V(T)$ is defined and belongs to $\overline{B_R}$, the closure of B_R, because $\overline{B_R}$ is closed, in effect to ∂B_R, we also have $\frac{d}{dt}\|V\|_F^2 |_{t=T} = f(T,V_T) \le -\delta < 0$ because $V(T) \in \partial B_R$. Thus we deduce that for $\varepsilon > 0$ and small enough, $V(T+\varepsilon) \in \overline{B_R}$ which contradicts the definition of T. Thus $T \notin R$ and $\overline{B_R}$ is stable.

Because f<0 on $\partial B_R, V(0) \in \partial B_R$ implies that $\forall t > 0, V(t) \in B_R$. Finally we consider the case $V(0) \in C\overline{B_R}$. Suppose that $\forall t > 0, V(t) \notin \overline{B_R}$, then $\forall t > 0, \frac{d}{dt}\|V\|_F^2 \le -2\delta$, thus $\|V(t)\|_F$ is monotonically decreasing and reaches the value of R in finite time when $V(t)$ reaches ∂B_R. This contradicts our assumption. Thus $\exists T > 0 \mid V(T) \in B_R$.

Proposition 1.1 : Let s and t be measured simple functions on X. for $E\varepsilon M$, define

$$\phi(E) = \int_E s\,d\mu \qquad (1)$$

Then ϕ is a measure on M.

$$\int_X (s+t)d\mu = \int_X s\,d\mu + \int_X t\,d\mu \qquad (2)$$

Proof : If s and if $E_1, E_2, \dots$ are disjoint members of M whose union is E, the countable additivity of μ shows that

$$\phi(E) = \sum_{i=1}^{n} \alpha_i \mu(A_i \cap E) = \sum_{i=1}^{n} \alpha_i \sum_{r=1}^{\infty} \mu(A_i \cap E_r)$$

$$= \sum_{r=1}^{\infty}\sum_{i=1}^{n} \alpha_i \mu(A_i \cap E_r) = \sum_{r=1}^{\infty} \phi(E_r)$$

Also, $\varphi(\phi)=0,$ so that φ is not identically ∞ .

Next, let s be as before, let $\beta_1,...,\beta_m$ be the distinct values of t,and let $B_j=\{x:t(x)=\beta_j\}$ If $E_{ij}=A_i\cap B_j,$ the $\int_{E_{ij}}(s+t)d\mu=(\alpha_i+\beta_j)\mu(E_{ij})$

and $\int_{E_{ij}} sd\mu+\int_{E_{ij}} td\mu=\alpha_i\mu(E_{ij})+\beta_j\mu(E_{ij})$

Thus (2) holds with E_{ij} in place of X . Since X is the disjoint union of the sets E_{ij} $(1\le i\le n, 1\le j\le m),$ the first half of our proposition implies that (2) holds.

Theorem 1.1: If K is a compact set in the plane whose complement is connected, if f is a continuous complex function on K which is holomorphic in the interior of , and if $\varepsilon>0,$ then there exists a polynomial P such that $|f(z)=P(z)|<\varepsilon$ for all $z\varepsilon K$. If the interior of K is empty, then part of the hypothesis is vacuously satisfied, and the conclusion holds for every $f\varepsilon C(K)$. Note that K need to be connected.

Proof: By Tietze's theorem, f can be extended to a continuous function in the plane, with compact support. We fix one such extension and denote it again by f . For any $\delta>0,$ let $\omega(\delta)$ be the supremum of the numbers $|f(z_2)-f(z_1)|$ Where z_1 and z_2 are subject to the condition $|z_2-z_1|\le\delta$. Since f is uniformly continous, we have $\lim_{\delta\to0}\omega(\delta)=0$ (1) From now on, δ will be fixed. We shall prove that there is a polynomial P such that

$$|f(z)-P(z)|<10,000\ \omega(\delta)\quad (z\varepsilon K)\qquad (2)$$

By (1), this proves the theorem. Our first objective is the construction of a function $\Phi\varepsilon C_c'(R^2),$ such that for all z

$$|f(z)-\Phi(z)|\le\omega(\delta),\qquad (3)$$

$$|(\partial\Phi)(z)|<\frac{2\omega(\delta)}{\delta},\qquad (4)$$

And

$$\Phi(z)=-\frac{1}{\pi}\iint_X\frac{(\partial\Phi)(\zeta)}{\zeta-z}d\zeta d\eta\qquad(\zeta=\xi+i\eta),\qquad (5)$$

Where X is the set of all points in the support of Φ whose distance from the complement of K does not δ . (Thus X contains no point which is "far within" K .) We construct Φ as the convolution of f with a smoothing function A. Put $a(r)=0$ if $r>\delta,$ put

$$a(r)=\frac{3}{\pi\delta^2}(1-\frac{r^2}{\delta^2})^2\qquad(0\le r\le\delta),\qquad (6)$$

And define

$$A(z)=a(|z|)\qquad (7)$$

For all complex z . It is clear that $A\varepsilon C_c'(R^2)$. We claim that

$$\iint_{R^s} A=1,\qquad (8)$$

$$\iint_{R^2}\partial A=0,\qquad (9)$$

$$\iint_{R^3}|\partial A|=\frac{24}{15\delta}<\frac{2}{\delta},\qquad (10)$$

The constants are so adjusted in (6) that (8) holds. (Compute the integral in polar coordinates), (9) holds simply because A has compact support. To compute (10), express ∂A in polar coordinates, and note that $\partial A/\partial\theta=0,$

$$\partial A/\partial r=-a',$$

Now define

$$\Phi(z)=\iint_{R^2} f(z-\zeta)Ad\xi d\eta=\iint_{R^2}A(z-\zeta)f(\zeta)d\xi d\eta\qquad (11)$$

Since f and A have compact support, so does Φ. Since

$$\Phi(z)-f(z)$$
$$=\iint_{R^2}[f(z-\zeta)-f(z)]A(\xi)d\xi d\eta\quad (12)$$

And $A(\zeta)=0$ if $|\zeta|>\delta,$ (3) follows from (8). The difference quotients of A converge boundedly to the corresponding partial derivatives, since $A\varepsilon C_c'(R^2)$. Hence the last expression in (11) may be differentiated under the integral sign, and we obtain

$$(\partial\Phi)(z) = \iint_{R^2} (\overline{\partial A})(z-\zeta) f(\zeta) d\xi d\eta$$

$$= \iint_{R^2} f(z-\zeta)(\partial A)(\zeta) d\xi d\eta$$

$$= \iint_{R^2} [f(z-\zeta) - f(z)](\partial A)(\zeta) d\xi d\eta \qquad (13)$$

The last equality depends on (9). Now (10) and (13) give (4). If we write (13) with Φ_x and Φ_y in place of $\partial\Phi,$ we see that Φ has continuous partial derivatives, if we can show that $\partial\Phi = 0$ in $G,$ where G is the set of all $z\varepsilon K$ whose distance from the complement of K exceeds $\delta.$ We shall do this by showing that

$$\Phi(z) = f(z) \qquad (z\varepsilon G); \qquad (14)$$

Note that $\partial f = 0$ in G , since f is holomorphic there. Now if $z\varepsilon G,$ then $z-\zeta$ is in the interior of K for all ζ with $|\zeta| < \delta.$ The mean value property for harmonic functions therefore gives, by the first equation in (11),

$$\Phi(z) = \int_0^{\delta} a(r) r dr \int_0^{2\pi} f(z - re^{i\theta}) d\theta$$

$$= 2\pi f(z) \int_0^{\delta} a(r) r dr = f(z) \iint_{R^2} A = f(z) \qquad (15)$$

For all $z\ \varepsilon\ G$, we have now proved (3), (4), and (5) The definition of X shows that X is compact and that X can be covered by finitely many open discs $D_1,\ldots,D_n,$ of radius $2\delta,$ whose centers are not in $K.$ Since $S^2 - K$ is connected, the center of each D_j can be joined to ∞ by a polygonal path in $S^2 - K$. It follows that each D_j contains a compact connected set $E_j,$ of diameter at least $2\delta,$ so that $S^2 - E_j$ is connected and so that $K \cap E_j = \phi.$ with $r = 2\delta$. There are functions $g_j \varepsilon H(S^2 - E_j)$ and constants b_j so that the inequalities.

$$|Q_j(\zeta,z)| < \frac{50}{\delta}, \qquad (16)$$

$$\left| Q_j(\zeta,z) - \frac{1}{z-\zeta} \right| < \frac{4{,}000\delta^2}{|z-\zeta|^2} \qquad (17)$$

Hold for $z \notin E_j$ and $\zeta \in D_j,$ if

$$Q_j(\zeta,z) = g_j(z) + (\zeta - b_j) g_j^2(z) \qquad (18)$$

Let Ω be the complement of $E_1 \cup \ldots \cup E_n.$ Then Ω is an open set which contains $K.$ Put $X_1 = X \cap D_1$ and $X_j = (X \cap D_j) - (X_1 \cup \ldots \cup X_{j-1}),$ for $2 \le j \le n,$

Define

$$R(\zeta,z) = Q_j(\zeta,z) \qquad (\zeta\varepsilon X_j, z\,\varepsilon\,\Omega) \qquad (19)$$

And

$$F(z) = \frac{1}{\pi}\iint_X (\partial\Phi)(\zeta) R(\zeta,z) d\zeta d\eta \qquad (20)$$

$$(z\ \varepsilon\ \Omega)$$

Since,

$$F(z) = \sum_{j=1} \frac{1}{\pi} \iint_{X_i} (\partial\Phi)(\zeta) Q_j(\zeta,z) d\xi d\eta, \qquad (21)$$

(18) shows that F is a finite linear combination of the functions g_j and g_j^2 . Hence $F\varepsilon H(\Omega).$ By (20), (4), and (5) we have

$$|F(z) - \Phi(z)| < \frac{2\omega(\delta)}{\pi\delta} \iint_X | R(\zeta,z) - \frac{1}{z-\zeta} | d\xi d\eta \quad (z\ \varepsilon\ \Omega) \quad (22)$$

Observe that the inequalities (16) and (17) are valid with R in place of Q_j if $\zeta\ \varepsilon\ X$ and $z\ \varepsilon\ \Omega.$ Now fix $z\ \varepsilon\ \Omega.$, put $\zeta = z + \rho e^{i\theta},$ and estimate the integrand in (22) by (16) if $\rho < 4\delta,$ by (17) if $4\delta \le \rho.$ The integral in (22) is then seen to be less than the sum of

$$2\pi \int_0^{4\delta} \left(\frac{50}{\delta} + \frac{1}{\rho} \right) \rho d\rho = 808\pi\delta \qquad (23)$$

And

$$2\pi \int_{4\delta}^{\infty} \frac{4{,}000\delta^2}{\rho^2} \rho d\rho = 2{,}000\pi\delta. \qquad (24)$$

Hence (22) yields

$$|F(z) - \Phi(z)| < 6{,}000\omega(\delta) \qquad (z\ \varepsilon\ \Omega) \qquad (25)$$

Since $F\ \varepsilon\ H(\Omega), K \subset \Omega,$ and $S^2 - K$ is connected, Runge's theorem shows that F can be uniformly approximated on K by polynomials. Hence (3) and (25) show that (2) can be satisfied. This completes the proof.

Lemma 1.0 : Suppose $f\varepsilon C_c^{'}(R^2)$, the space of all continuously differentiable functions in the plane, with compact support. Put

$$\partial=\frac{1}{2}\left(\frac{\partial}{\partial x}+i\frac{\partial}{\partial y}\right) \qquad (1)$$

Then the following "Cauchy formula" holds:

$$f(z)=-\frac{1}{\pi}\iint_{R^2}\frac{(\partial f)(\zeta)}{\zeta-z}d\xi d\eta$$

$$(\zeta=\xi+i\eta) \qquad (2)$$

Proof: This may be deduced from Green's theorem. However, here is a simple direct proof:

Put $\varphi(r,\theta)=f(z+re^{i\theta}),\ r>0,\ \theta$ real

If $\zeta=z+re^{i\theta}$, the chain rule gives

$$(\partial f)(\zeta)=\frac{1}{2}e^{i\theta}\left[\frac{\partial}{\partial r}+\frac{i}{r}\frac{\partial}{\partial\theta}\right]\varphi(r,\theta) \qquad (3)$$

The right side of (2) is therefore equal to the limit, as $\varepsilon\to 0$, of

$$-\frac{1}{2}\int_{\varepsilon}^{\infty}\int_{0}^{2\pi}\left(\frac{\partial\varphi}{\partial r}+\frac{i}{r}\frac{\partial\varphi}{\partial\theta}\right)d\theta dr \qquad (4)$$

For each $r>0,\varphi$ is periodic in θ, with period 2π. The integral of $\partial\varphi/\partial\theta$ is therefore 0, and (4) becomes

$$-\frac{1}{2\pi}\int_{0}^{2\pi}d\theta\int_{\varepsilon}^{\infty}\frac{\partial\varphi}{\partial r}dr=\frac{1}{2\pi}\int_{0}^{2\pi}\varphi(\varepsilon,\theta)d\theta \qquad (5)$$

As $\varepsilon\to 0,\ \varphi(\varepsilon,\theta)\to f(z)$ uniformly. This gives (2)

If $X^{\alpha}\in a$ and $X^{\beta}\in k[X_1,...X_n]$, then $X^{\alpha}X^{\beta}=X^{\alpha+\beta}\in a$, and so A satisfies the condition $(*)$. Conversely,

$$\Big(\sum_{\alpha\in A}c_{\alpha}X^{\alpha}\Big)\Big(\sum_{\beta\in\square^{n}}d_{\beta}X^{\beta}\Big)=\sum_{\alpha,\beta}c_{\alpha}d_{\beta}X^{\alpha+\beta} \qquad (finite\,sums),$$

and so if A satisfies $(*)$, then the subspace generated by the monomials $X^{\alpha},\alpha\in a$, is an ideal. The proposition gives a classification of the monomial ideals in $k[X_1,...X_n]$: they are in one to one correspondence with the subsets A of $\square^{n}$ satisfying $(*)$. For example, the monomial ideals in $k[X]$ are exactly the ideals $(X^{n}),\ n\geq 1$, and the zero ideal (corresponding to the empty set A). We write $\left\langle X^{\alpha}\mid\alpha\in A\right\rangle$ for the ideal corresponding to A (subspace generated by the $X^{\alpha},\alpha\in a$).

LEMMA 1.1. Let S be a subset of $\square^{n}$. The the ideal a generated by $X^{\alpha},\alpha\in S$ is the monomial ideal corresponding to

$$A\overset{df}{=}\left\{\beta\in\square^{n}\mid\beta-\alpha\in\square^{n},\quad some\ \alpha\in S\right\}$$

Thus, a monomial is in a if and only if it is divisible by one of the $X^{\alpha},\alpha\in| S$

PROOF. Clearly A satisfies $(*)$, and $a\subset\left\langle X^{\beta}\mid\beta\in A\right\rangle$. Conversely, if $\beta\in A$, then $\beta-\alpha\in\square^{n}$ for some $\alpha\in S$, and $X^{\beta}=X^{\alpha}X^{\beta-\alpha}\in a$. The last statement follows from the fact that $X^{\alpha}\mid X^{\beta}\Leftrightarrow\beta-\alpha\in\square^{n}$. Let $A\subset\square^{n}$ satisfy $(*)$. From the geometry of A, it is clear that there is a finite set of elements $S=\left\{\alpha_1,...\alpha_s\right\}$ of A such that $A=\left\{\beta\in\square^{n}\mid\beta-\alpha_i\in\square^{2},\ some\ \alpha_i\in S\right\}$ (The α_i's are the corners of A) Moreover, $a\overset{df}{=}\left\langle X^{\alpha}\mid\alpha\in A\right\rangle$ is generated by the monomials $X^{\alpha_i},\alpha_i\in S$.

DEFINITION 1.0. For a nonzero ideal a in $k\left[X_1,...,X_n\right]$, we let $(LT(a))$ be the ideal generated by

$\left\{LT(f)\mid f\in a\right\}$

LEMMA 1.2 Let a be a nonzero ideal in $k\left[X_1,...,X_n\right]$; then $(LT(a))$ is a monomial ideal, and it equals $(LT(g_1),...,LT(g_n))$ for some $g_1,...,g_n\in a$.

PROOF. Since $(LT(a))$ can also be described as the ideal generated by the leading monomials (rather than the leading terms) of elements of a.

THEOREM 1.2. Every *ideal* a in $k\left[X_1,...,X_n\right]$ is finitely generated; more precisely, $a=(g_1,...,g_s)$ where $g_1,...,g_s$ are any elements of a whose leading terms generate $LT(a)$

PROOF. Let $f \in a$. On applying the division algorithm, we find $f = a_1g_1 + ... + a_sg_s + r, \quad a_i, r \in k\left[X_1,...,X_n\right]$, where either $r = 0$ or no monomial occurring in it is divisible by any $LT(g_i)$. But $r = f - \sum a_i g_i \in a$, and therefore $LT(r) \in LT(a) = (LT(g_1),...,LT(g_s))$, implies that every monomial occurring in r is divisible by one in $LT(g_i)$. Thus $r = 0$, and $g \in (g_1,...,g_s)$.

DEFINITION 1.1. A finite subset $S = \left\{g_1,|...,g_s\right\}$ of an ideal a is a standard ($(Gr\ddot{o}bner)$ bases for a if $(LT(g_1),...,LT(g_s)) = LT(a)$. In other words, S is a standard basis if the leading term of every element of a is divisible by at least one of the leading terms of the g_i.

THEOREM 1.3 *The ring $k[X_1,...,X_n]$ is Noetherian i.e., every ideal is finitely generated.*

PROOF. For $n = 1$, $k[X]$ is a principal ideal domain, which means that every ideal is generated by single element. We shall prove the theorem by induction on n. Note that the obvious map $k[X_1,...X_{n-1}][X_n] \to k[X_1,...X_n]$ is an isomorphism – this simply says that every polynomial f in n variables $X_1,...X_n$ can be expressed uniquely as a polynomial in X_n with coefficients in $k[X_1,...,X_n]$:

$$f(X_1,...X_n) = a_0(X_1,...X_{n-1})X_n^r + ... + a_r(X_1,...X_{n-1})$$

Thus the next lemma will complete the proof

LEMMA 1.3. If A is Noetherian, then so also is $A[X]$

PROOF. For a polynomial

$$f(X) = a_0X^r + a_1X^{r-1} + ... + a_r, \quad a_i \in A, \quad a_0 \neq 0,$$

r is called the degree of f, and a_0 is its leading coefficient. We call 0 the leading coefficient of the polynomial 0. Let a be an ideal in $A[X]$. The leading coefficients of the polynomials in a form an ideal a' in A, and since A is Noetherian, a' will be finitely generated. Let $g_1,...,g_m$ be elements of a whose leading coefficients generate a', and let r be the maximum degree of g_i. Now let $f \in a,$ and suppose f has degree $s > r$, say, $f = aX^s + ...$ Then $a \in a'$, and so we can write

$a = \sum b_i a_i, \qquad b_i \in A,$

$a_i = leading\ coefficient\ of\ g_i$

Now

$f - \sum b_i g_i X^{s-r_i}, \quad r_i = \deg(g_i),$ has degree $< \deg(f)$. By continuing in this way, we find that $f \equiv f_t \quad \mod(g_1,...g_m)$ With f_t a polynomial of degree $t < r$. For each $d < r$, let a_d be the subset of A consisting of 0 and the leading coefficients of all polynomials in a of degree d; it is again an ideal in A. Let $g_{d,1},...,g_{d,m_d}$ be polynomials of degree d whose leading coefficients generate a_d. Then the same argument as above shows that any polynomial f_d in a of degree d can be written $f_d \equiv f_{d-1} \quad \mod(g_{d,1},...g_{d,m_d})$ With f_{d-1} of degree $\leq d-1$. On applying this remark repeatedly we find that $f_t \in (g_{r-1,1},...g_{r-1,m_{r-1}},...g_{0,1},...g_{0,m_0})$ Hence

$$f_t \in (g_1,...g_m g_{r-1,1},...g_{r-1,m_{r-1}},..., g_{0,1},..., g_{0,m_0})$$

and so the polynomials $g_1,..., g_{0,m_0}$ generate a

One of the great successes of category theory in computer science has been the development of a "unified theory" of the constructions underlying denotational semantics. In the untyped λ-calculus, any term may appear in the function position of an application. This means that a model D of the λ-calculus must have the property that given a term t whose interpretation is $d \in D,$ Also, the interpretation of a functional abstraction like $\lambda x . x$ is most conveniently defined as a function from $D to D$, which must then be regarded as an element of D. Let $\psi : \left[D \to D\right] \to D$ be the function that picks out elements of D to represent elements of $\left[D \to D\right]$ and $\phi : D \to \left[D \to D\right]$ be the function that maps elements of D to functions of D. Since $\psi(f)$ is intended to represent the function f as an element of *D*, it makes sense to require that $\phi(\psi(f)) = f$, that is, $\psi o \psi = id_{[D\to D]}$ Furthermore, we often want to view every element of *D* as representing some function from *D to D* and require that elements representing the same function be equal – that is

$\psi(\varphi(d)) = d$

or

$\psi o \phi = id_D$

The latter condition is called extensionality. These conditions together imply that $\phi and \psi$ are inverses--- that is, *D* is isomorphic to the space of functions from *D to D* that can be the interpretations of functional abstractions: $D \cong [D \to D]$ .Let us suppose we are working with the untyped $\lambda - calculus$, we need a solution ot the equation $D \cong A + [D \to D],$ where A is some predetermined domain containing interpretations for elements of *C*. Each element of *D* corresponds to either an element of *A* or an element of $[D \to D]$, with a tag. This equation can be solved by finding least fixed points of the function $F(X) = A + [X \to X]$ from domains to domains --- that is, finding domains *X* such that $X \cong A + [X \to X]$, and such that for any domain *Y* also satisfying this equation, there is an embedding of *X* to *Y* --- a pair of maps

$$X \overset{f}{\underset{f^R}{\square}} Y$$

Such that

$f^R o f = id_X$

$f o f^R \subseteq id_Y$

Where $f \subseteq g$ means that $f\ approximates\ g$ in some ordering representing their information content. The key shift of perspective from the domain-theoretic to the more general category-theoretic approach lies in considering *F* not as a function on domains, but as a *functor* on a category of domains. Instead of a least fixed point of the function, *F*.

Definition 1.3: Let ***K*** be a category and $F : K \to K$ as a functor. A fixed point of *F* is a pair (A,a), where A is a ***K-object*** and $a : F(A) \to A$ is an isomorphism. A prefixed point of F is a pair (A,a), where A is a ***K-object*** and a is any arrow from F(A) to A

Definition 1.4 : An $\omega - chain$ in a category ***K*** is a diagram of the following form:

$$\Delta = D_o \xrightarrow{f_o} D_1 \xrightarrow{f_1} D_2 \xrightarrow{f_2} .....$$

Recall that a cocone μ of an $\omega - chain$ Δ is a *K*-object *X* and a collection of K *–arrows* $\{\mu_i : D_i \to X \mid i \geq 0\}$ such that $\mu_i = \mu_{i+1} o\, f_i$ for all $i \geq 0$. We sometimes write $\mu : \Delta \to X$ as a reminder of the arrangement of $\mu's$ components Similarly, a colimit $\mu : \Delta \to X$ is a cocone with the property that if $\nu : \Delta \to X'$ is also a cocone then there exists a unique mediating arrow $k : X \to X'$ such that for all $i \geq 0,,\ \nu_i = k\, o\, \mu_i$. Colimits of $\omega - chains$ are sometimes referred to as $\omega - co\lim its$. Dually, an $\omega^{op} - chain$ in ***K*** is a diagram of the following form:

$$\Delta = D_o \xleftarrow{f_o} D_1 \xleftarrow{f_1} D_2 \xleftarrow{f_2} .....$$

A cone $\mu : X \to \Delta$ of an $\omega^{op} - chain$ Δ is a ***K***-object X and a collection of **K**-arrows $\{\mu_i : D_i \mid i \geq 0\}$ such that for all $i \geq 0,\ \mu_i = f_i\, o\, \mu_{i+1}$. An ω^{op} -limit of an $\omega^{op} - chain$ Δ is a cone $\mu : X \to \Delta$ with the property that if $\nu : X' \to \Delta$ is also a cone, then there exists a unique mediating arrow $k : X' \to X$ such that for all $i \geq 0, \mu_i\, o\, k = \nu_i$. We write $\perp_k$ (or just $\perp$) for the distinguish initial object of ***K,*** when it has one, and $\perp \to A$ for the unique arrow from $\perp$ to each ***K***-object A. It is also convenient to write $\Delta^- = D_1 \xrightarrow{f_1} D_2 \xrightarrow{f_2} .....$ to denote all of Δ except D_o and f_0. By analogy, μ^- is $\{\mu_i \mid i \geq 1\}$. For the images of Δ and μ under ***F*** we write

$$F(\Delta) = F(D_o) \xrightarrow{F(f_o)} F(D_1) \xrightarrow{F(f_1)} F(D_2) \xrightarrow{F(f_2)} .....$$

and $F(\mu) = \{F(\mu_i) \mid i \geq 0\}$

We write F^i for the ***i***-fold iterated composition of ***F*** – that is, $F^o(f) = f, F^1(f) = F(f), F^2(f) = F(F(f))$,etc. With these definitions we can state that every monitonic function on a complete lattice has a least fixed point:

Lemma 1.4. Let ***K*** be a category with initial object $\perp$ and let $F : K \to K$ be a functor. Define the $\omega - chain \Delta$ by

$$\Delta = \perp \xrightarrow{!\perp \to F(\perp)} F(\perp) \xrightarrow{F(!\perp \to F(\perp))} F^2(\perp) \xrightarrow{F^2(!\perp \to F(\perp))} \dots\dots\dots$$

If both $\mu : \Delta \to D$ and $F(\mu) : F(\Delta) \to F(D)$ are colimits, then (D,d) is an intial F-algebra, where $d : F(D) \to D$ is the mediating arrow from $F(\mu)$ to the cocone μ^{-}

Theorem 1.4 Let a DAG G given in which each node is a random variable, and let a discrete conditional probability distribution of each node given values of its parents in G be specified. Then the product of these conditional distributions yields a joint probability distribution P of the variables, and (G,P) satisfies the Markov condition.

Proof. Order the nodes according to an ancestral ordering. Let $X_1, X_2, \dots\dots\dots X_n$ be the resultant ordering. Next define.

$$P(x_1, x_2, \dots . x_n) = P(x_n \mid pa_n) P(x_{n-1} \mid Pa_{n-1}) \dots$$
$$\dots P(x_2 \mid pa_2) P(x_1 \mid pa_1),$$

Where PA_i is the set of parents of X_i of in G and $P(x_i \mid pa_i)$ is the specified conditional probability distribution. First we show this does indeed yield a joint probability distribution. Clearly, $0 \le P(x_1, x_2, \dots x_n) \le 1$ for all values of the variables. Therefore, to show we have a joint distribution, as the variables range through all their possible values, is equal to one. To that end, Specified conditional distributions are the conditional distributions they notationally represent in the joint distribution. Finally, we show the Markov condition is satisfied. To do this, we need show for $1 \le k \le n$ that whenever

$$P(pa_k) \ne 0, if \;\; P(nd_k \mid pa_k) \ne 0$$
$$and \;\; P(x_k \mid pa_k) \ne 0$$
$$then \; P(x_k \mid nd_k, pa_k) = P(x_k \mid pa_k),$$

Where ND_k is the set of nondescendents of X_k of in G. Since $PA_k \subseteq ND_k$, we need only show $P(x_k \mid nd_k) = P(x_k \mid pa_k)$. First for a given k, order the nodes so that all and only nondescendents of X_k precede X_k in the ordering. Note that this ordering depends on k, whereas the ordering in the first part of the proof does not. Clearly then

$$ND_k = \{X_1, X_2, \dots . X_{k-1}\}$$
$$Let$$
$$D_k = \{X_{k+1}, X_{k+2}, \dots . X_n\}$$

follows $\sum_{d_k}$

We define the m^{th} *cyclotomic field to be the field* $Q[x]/(\Phi_m(x))$ Where $\Phi_m(x)$ is the m^{th} cyclotomic polynomial. $Q[x]/(\Phi_m(x))$ $\Phi_m(x)$ *has degree* $\varphi(m)$ *over* Q *since* $\Phi_m(x)$ has degree $\varphi(m)$. *The roots of* $\Phi_m(x)$ are just the primitive m^{th} roots of unity, so the complex embeddings of $Q[x]/(\Phi_m(x))$ *are simply the* $\varphi(m)$ *maps*

$$\sigma_k : Q[x]/(\Phi_m(x)) \mapsto C,$$
$$1 \le k \prec m, (k,m) = 1, \quad where$$
$$\sigma_k(x) = \xi_m^k,$$

ξ_m being our fixed choice of primitive m^{th} root of unity. Note that $\xi_m^k \in Q(\xi_m)$ for every k; it follows that $Q(\xi_m) = Q(\xi_m^k)$ for all k relatively prime to m. In particular, the images of the σ_i coincide, so $Q[x]/(\Phi_m(x))$ *is Galois over* Q. *This means that we can write* $Q(\xi_m)$ *for* $Q[x]/(\Phi_m(x))$ *without much fear of ambiguity; we will do so from now on, the identification being* $\xi_m \mapsto x$. *One advantage of this is that one can easily talk about cyclotomic fields being extensions of one another,or intersections or compositums; all of these things take place considering them as subfield of* C. We now investigate some basic properties of cyclotomic fields. The first issue is whether or not they are all distinct; to determine this, we need to know which roots of unity lie in $Q(\xi_m)$ .Note, for example, that if m is odd, then $-\xi_m$ is a $2m^{th}$ root of unity. We will show that this is the only way in which one can obtain any non-m^{th} roots of unity.

LEMMA 1.5 If m divides n, then $Q(\xi_m)$ *is contained in* $Q(\xi_n)$

PROOF. Since $\xi^{n/m} = \xi_m$, *we have* $\xi_m \in Q(\xi_n)$, *so the result is clear*

LEMMA 1.6 If m and n are relatively prime, then

$$Q(\xi_m, \xi_n) = Q(\xi_{nm})$$

and

$$Q(\xi_m) \cap Q(\xi_n) = Q$$

(Recall the $Q(\xi_m, \xi_n)$ is the compositum of $Q(\xi_m)$ *and* $Q(\xi_n)$)

PROOF. One checks easily that $\xi_m \xi_n$ is a primitive mn^{th} root of unity, so that

$$Q(\xi_{mn}) \subseteq Q(\xi_m, \xi_n)$$
$$[Q(\xi_m, \xi_n) : Q] \le [Q(\xi_m) : Q][Q(\xi_n : Q]$$
$$= \varphi(m)\varphi(n) = \varphi(mn);$$

Since $[Q(\xi_{mn}) : Q] = \varphi(mn);$ this implies that $Q(\xi_m, \xi_n) = Q(\xi_{nm})$ We know that $Q(\xi_m, \xi_n)$ has degree $\varphi(mn)$ over Q , so we must have

$$[Q(\xi_m, \xi_n) : Q(\xi_m)] = \varphi(n)$$

and

$$[Q(\xi_m, \xi_n) : Q(\xi_m)] = \varphi(m)$$

$$[Q(\xi_m) : Q(\xi_m) \cap Q(\xi_n)] \ge \varphi(m)$$

And thus that $Q(\xi_m) \cap Q(\xi_n) = Q$

PROPOSITION 1.2 For any m and n

$$Q(\xi_m, \xi_n) = Q(\xi_{[m,n]})$$

And

$$Q(\xi_m) \cap Q(\xi_n) = Q(\xi_{(m,n)});$$

here $[m, n]$ and (m, n) denote the least common multiple and the greatest common divisor of m and n, respectively.

PROOF. Write $m = p_1^{e_1} ......p_k^{e_k}$ *and* $p_1^{f_1} ....p_k^{f_k}$ where the p_i are distinct primes. (We allow e_i *or* f_i to be zero)

$$Q(\xi_m) = Q(\xi_{p_1^{e_1}})Q(\xi_{p_2^{e_2}})...Q(\xi_{p_k^{e_k}})$$

and

$$Q(\xi_n) = Q(\xi_{p_1^{f_1}})Q(\xi_{p_2^{f_2}})...Q(\xi_{p_k^{f_k}})$$

Thus

$$\begin{aligned} Q(\xi_m, \xi_n) &= Q(\xi_{p_1^{e_1}})........Q(\xi_{p_2^{e_k}})Q(\xi_{p_1^{f_1}})...Q(\xi_{p_k^{f_k}}) \\ &= Q(\xi_{p_1^{e_1}})Q(\xi_{p_1^{f_1}})...Q(\xi_{p_k^{e_k}})Q(\xi_{p_k^{f_k}}) \\ &= Q(\xi_{p_1^{\max(e_1, f_1)}}).......Q(\xi_{p_1^{\max(e_k, f_k)}}) \\ &= Q(\xi_{p_1^{\max(e_1, f_1)}........p_1^{\max(e_k, f_k)}}) \\ &= Q(\xi_{[m,n]}); \end{aligned}$$

An entirely similar computation shows that $Q(\xi_m) \cap Q(\xi_n) = Q(\xi_{(m,n)})$

Mutual information measures the information transferred when x_i is sent and y_i is received, and is defined as

$$I(x_i, y_i) = \log_2 \frac{P({x_i}/{y_i})}{P(x_i)} \; bits \qquad (1)$$

In a noise-free channel, **each** y_i is uniquely connected to the corresponding x_i , and so they constitute an input –output pair (x_i, y_i) for which

$$P({x_i}/{y_j}) = 1 \; and \; I(x_i, y_j) = \log_2 \frac{1}{P(x_i)} \quad \text{bits;}$$

that is, the transferred information is equal to the self-information that corresponds to the input x_i In a very noisy channel, the output y_i and input x_i would be completely uncorrelated, and so $P({x_i}/{y_j}) = P(x_i)$ and also $I(x_i, y_j) = 0;$ that is, there is no transference of information. In general, a given channel will operate between these two extremes. The mutual information is defined between the input and the output of a given channel. An average of the calculation of the mutual information for all input-output pairs of a given channel is the average mutual information:

$$I(X,Y) = \sum_{i.j} P(x_i, y_j) I(x_i, y_j) = \sum_{i.j} P(x_i, y_j) \log_2 \left[\frac{P({x_i}/{y_j})}{P(x_i)} \right]$$

bits per symbol . This calculation is done over the input and output alphabets. The average mutual information. The following expressions are useful for modifying the mutual information expression:

$$P(x_i,y_j)=P({}^{x_i}\!/_{y_j})P(y_j)=P({}^{y_j}\!/_{x_i})P(x_i)$$

$$P(y_j)=\sum_i P({}^{y_j}\!/_{x_i})P(x_i)$$

$$P(x_i)=\sum_i P({}^{x_i}\!/_{y_j})P(y_j)$$

Then

$$I(X,Y)=\sum_{i.j}P(x_i,y_j)$$

$$=\sum_{i.j}P(x_i,y_j)\log_2\left[\frac{1}{P(x_i)}\right]$$

$$-\sum_{i.j}P(x_i,y_j)\log_2\left[\frac{1}{P({}^{x_i}\!/_{y_j})}\right]$$

$$\sum_{i.j}P(x_i,y_j)\log_2\left[\frac{1}{P(x_i)}\right]$$

$$=\sum_i\left[P({}^{x_i}\!/_{y_j})P(y_j)\right]\log_2\frac{1}{P(x_i)}$$

$$\sum_i P(x_i)\log_2\frac{1}{P(x_i)}=H(X)$$

$$I(X,Y)=H(X)-H({}^{X}\!/_{Y})$$

Where $H({}^{X}\!/_{Y})=\sum_{i,j}P(x_i,y_j)\log_2\dfrac{1}{P({}^{x_i}\!/_{y_j})}$

is usually called the equivocation. In a sense, the equivocation can be seen as the information lost in the noisy channel, and is a function of the backward conditional probability. The observation of an output symbol y_j provides $H(X)-H({}^{X}\!/_{Y})$ bits of information. This difference is the mutual information of the channel. *Mutual Information: Properties* Since

$$P({}^{x_i}\!/_{y_j})P(y_j)=P({}^{y_j}\!/_{x_i})P(x_i)$$

The mutual information fits the condition

$$I(X,Y)=I(Y,X)$$

And by interchanging input and output it is also true that

$$I(X,Y)=H(Y)-H({}^{Y}\!/_{X})$$

Where

$$H(Y)=\sum_j P(y_j)\log_2\frac{1}{P(y_j)}$$

This last entropy is usually called the noise entropy. Thus, the information transferred through the channel is the difference between the output entropy and the noise entropy. Alternatively, it can be said that the channel mutual information is the difference between the number of bits needed for determining a given input symbol before knowing the corresponding output symbol, and the number of bits needed for determining a given input symbol after knowing the corresponding output symbol

$$I(X,Y)=H(X)-H({}^{X}\!/_{Y})$$

As the channel mutual information expression is a difference between two quantities, it seems that this parameter can adopt negative values. However, and is spite of the fact that for some $y_j, H(X/y_j)$ can be larger than $H(X)$, this is not possible for the average value calculated over all the outputs:

$$\sum_{i,j}P(x_i,y_j)\log_2\frac{P({}^{x_i}\!/_{y_j})}{P(x_i)}=\sum_{i,j}P(x_i,y_j)\log_2\frac{P(x_i,y_j)}{P(x_i)P(y_j)}$$

Then

$$-I(X,Y)=\sum_{i,j}P(x_i,y_j)\frac{P(x_i)P(y_j)}{P(x_i,y_j)}\leq 0$$

Because this expression is of the form

$$\sum_{i=1}^{M}P_i\log_2(\frac{Q_i}{P_i})\leq 0$$

The above expression can be applied due to the factor $P(x_i)P(y_j)$, which is the product of two probabilities, so that it behaves as the quantity Q_i, which in this expression is a dummy variable that fits the condition $\sum_i Q_i\leq 1$. It can be concluded that the average mutual information is a non-negative number. It can also be equal to zero, when the input and the output are independent of each other. A related entropy called the joint entropy is defined as

$$H(X,Y)=\sum_{i,j}P(x_i,y_j)\log_2\frac{1}{P(x_i,y_j)}$$

$$=\sum_{i,j}P(x_i,y_j)\log_2\frac{P(x_i)P(y_j)}{P(x_i,y_j)}$$

$$+\sum_{i,j}P(x_i,y_j)\log_2\frac{1}{P(x_i)P(y_j)}$$

Theorem 1.5: Entropies of the binary erasure channel (BEC) The BEC is defined with an alphabet of two inputs and three outputs, with symbol probabilities.

$P(x_1)=\alpha\ \ and\ \ P(x_2)=1-\alpha,$ and transition probabilities

$$P({}^{y_3}\!/\!{}_{x_2}) = 1 - p \;\; and \;\; P({}^{y_2}\!/\!{}_{x_1}) = 0,$$

$$and \;\; P({}^{y_3}\!/\!{}_{x_1}) = 0$$

$$and \;\; P({}^{y_1}\!/\!{}_{x_2}) = p$$

$$and \;\; P({}^{y_3}\!/\!{}_{x_2}) = 1 - p$$

Lemma 1.7. Given an arbitrary restricted time-discrete, amplitude-continuous channel whose restrictions are determined by sets F_n and whose density functions exhibit no dependence on the state s, let n be a fixed positive integer, and $p(x)$ an arbitrary probability density function on Euclidean *n*-space. $p(y|x)$ for the density $p_n(y_1,...,y_n | x_1,...x_n)$ and $F \; for \; F_n$. For any real number a, let

$$A = \left\{ (x, y) : \log \frac{p(y|x)}{p(y)} > a \right\} \qquad (1)$$

Then for each positive integer u, there is a code (u, n, λ) such that

$$\lambda \le ue^{-a} + P\{(X,Y) \notin A\} + P\{X \notin F\} \qquad (2)$$

Where

$$P\{(X,Y) \in A\} = \int_A ... \int p(x,y) dxdy, \qquad p(x,y) = p(x) p(y|x)$$

and

$$P\{X \in F\} = \int_F ... \int p(x) dx$$

Proof: A sequence $x^{(1)} \in F$ *such that*

$$P\left\{ Y \in A_{x^1} \mid X = x^{(1)} \right\} \ge 1 - \varepsilon$$

$$where \;\; A_x = \{ y : (x, y) \varepsilon A \};$$

Choose the decoding set B_1 to be $A_{x^{(1)}}$. Having chosen $x^{(1)}, ..., x^{(k-1)}$ and $B_1, ..., B_{k-1}$, select $x^k \in F$ such that

$$P\left\{ Y \in A_{x^{(k)}} - \bigcup_{i=1}^{k-1} B_i \mid X = x^{(k)} \right\} \ge 1 - \varepsilon;$$

Set $B_k = A_{x^{(k)}} - \bigcup_{i=1}^{k-1} B_i$, If the process does not terminate in a finite number of steps, then the sequences $x^{(i)}$ and decoding sets B_i, $i = 1, 2, ..., u$, form the desired code. Thus assume that the process terminates after t steps. (Conceivably $t = 0$). We will show $t \ge u$ by showing that $\varepsilon \le te^{-a} + P\{(X,Y) \notin A\} + P\{X \notin F\}$. We proceed as follows.

Let

$$B = \bigcup_{j=1}^{t} B_j . \quad (If \;\; t = 0, \; take \;\; B = \phi). \;\; Then$$

$$P\{(X,Y) \in A\} = \int\limits_{(x,y) \in A} p(x,y) dx\, dy$$

$$= \int\limits_{x} p(x) \int\limits_{y \in A_x} p(y|x) dy\, dx$$

$$= \int\limits_{x} p(x) \int\limits_{y \in B \cap A_x} p(y|x) dy\, dx + \int\limits_{x} p(x)$$

E. Algorithms

Ideals. Let A be a ring. Recall that an *ideal a* in A is a subset such that a is subgroup of A regarded as a group under addition;

$$a \in a, r \in A \Rightarrow ra \in A$$

The ideal generated by a subset S of A is the intersection of all ideals A containing a ----- it is easy to verify that this is in fact an ideal, and that it consist of all finite sums of the form $\sum r_i s_i$ with $r_i \in A, s_i \in S$. When $S = \{s_1, ..., s_m\}$, we shall write $(s_1, ..., s_m)$ for the ideal it generates.

Let a and b be ideals in A. The set $\{a + b | a \in a, b \in b\}$ is an ideal, denoted by $a + b$. The ideal generated by $\{ab | a \in a, b \in b\}$ is denoted by ab. Note that $ab \subset a \cap b$. Clearly ab consists of all finite sums $\sum a_i b_i$ with $a_i \in a$ and $b_i \in b$, and if $a = (a_1, ..., a_m)$ and $b = (b_1, ..., b_n)$, then $ab = (a_1 b_1, ..., a_i b_j, ..., a_m b_n)$ .Let a be an ideal of A. The set of cosets of a in A forms a ring A / a , and $a \mapsto a + a$ is a homomorphism $\phi : A \mapsto A / a$. The map $b \mapsto \phi^{-1}(b)$ is a one to one correspondence between the ideals of A / a and the ideals of A containing a An ideal p if *prime* if $p \ne A$ and $ab \in p \Rightarrow a \in p$ or $b \in p$. Thus p is prime if and only if A / p is nonzero and has the property that $ab = 0, \quad b \ne 0 \Rightarrow a = 0,$ i.e., A / p is an integral domain. An ideal m is *maximal* if $m \ne\!| A$ and there does not exist an ideal n contained strictly between m and A. Thus m is maximal if and only if A / m has no proper nonzero ideals, and so is a field. Note that m maximal $\Rightarrow$ m prime. The ideals of $A \times B$ are all of the form $a \times b$, with a and b ideals in A and B . To see this, note that if c is an ideal in $A \times B$ and $(a,b) \in c$, then $(a,0) = (a,b)(1,0) \in c$ and $(0,b) = (a,b)(0,1) \in c$. This shows that $c = a \times b$ with

$$a = \{a \mid (a,b) \in c \ \ some \ \ b \in b\}$$

and

$$b = \{b \mid (a,b) \in c \ \ some \ \ a \in a\}$$

Let A be a ring. An A-algebra is a ring B together with a homomorphism $i_B : A \to B$. A *homomorphism of* A -algebra $B \to C$ is a homomorphism of rings $\varphi : B \to C$ such that $\varphi(i_B(a)) = i_C(a)$ for all $a \in A$. An A-algebra B is said to be *finitely generated* (or of *finite-type* over A) if there exist elements $x_1, \ldots, x_n \in B$ such that every element of B can be expressed as a polynomial in the x_i with coefficients in $i(A)$, i.e., such that the homomorphism $A[X_1, \ldots, X_n] \to B$ sending X_i to x_i is surjective. A ring homomorphism $A \to B$ is *finite*, and B is finitely generated as an A-module. Let k be a field, and let A be a k -algebra. If $1 \neq 0$ in A, then the map $k \to A$ is injective, we can identify k with its image, i.e., we can regard k as a subring of A. If 1=0 in a ring R, the R is the zero ring, i.e., $R = \{0\}$. **Polynomial rings.** Let k be a field. A *monomial* in $X_1, \ldots, X_n$ is an expression of the form $X_1^{a_1} \ldots X_n^{a_n}$, $a_j \in N$. The *total degree* of the monomial is $\sum a_i$. We sometimes abbreviate it by X^{α}, $\alpha = (a_1, \ldots, a_n) \in \square^{n}$. The elements of the polynomial ring $k[X_1, \ldots, X_n]$ are finite sums $\sum c_{a_1 \ldots a_n} X_1^{a_1} \ldots X_n^{a_n}$, $c_{a_1 \ldots a_n} \in k, \quad a_j \in \square$ With the obvious notions of equality, addition and multiplication. Thus the monomials from basis for $k[X_1, \ldots, X_n]$ as a k -vector space. The ring $k[X_1, \ldots, X_n]$ is an integral domain, and the only units in it are the nonzero constant polynomials. A polynomial $f(X_1, \ldots, X_n)$ is *irreducible* if it is nonconstant and has only the obvious factorizations, i.e., $f = gh \Rightarrow g$ or h is constant. **Division in** $k[X]$. The division algorithm allows us to divide a nonzero polynomial into another: let f and g be polynomials in $k[X]$ with $g \neq 0$; then there exist unique polynomials $q, r \in k[X]$ such that $f = qg + r$ with either $r = 0$ or $\deg r < \deg g$. Moreover, there is an algorithm for deciding whether $f \in (g)$, namely, find r and check whether it is zero. Moreover, the Euclidean algorithm allows to pass from finite set of generators for an ideal in $k[X]$ to a single generator by successively replacing each pair of generators with their greatest common divisor.

(Pure) ***lexicographic*** *ordering (lex)***.** Here monomials are ordered by lexicographic(dictionary) order. More precisely, let $\alpha = (a_1, \ldots a_n)$ and $\beta = (b_1, \ldots b_n)$ be two elements of $\square^{n}$; then $\alpha > \beta \ and \ X^{\alpha} > X^{\beta}$ (lexicographic ordering) if, in the vector difference $\alpha - \beta \in \square$, the left most nonzero entry is positive. For example, $XY^2 > Y^3Z^4; \quad X^3Y^2Z^4 > X^3Y^2Z$. Note that this isn't quite how the dictionary would order them: it would put *XXXYYZZZZ* after *XXXYYZ*. *Graded reverse lexicographic order (grevlex).* Here monomials are ordered by total degree, with ties broken by reverse lexicographic ordering. Thus, $\alpha > \beta$ if $\sum a_i > \sum b_i$, or $\sum a_i = \sum b_i$ and in $\alpha - \beta$ the right most nonzero entry is negative. For example:

$$X^4Y^4Z^7 > X^5Y^5Z^4 \ \ \textit{(total degree greater)}$$

$$XY^5Z^2 > X^4YZ^3, \quad X^5YZ > X^4YZ^2.$$

Orderings on $k[X_1, \ldots X_n]$**.** Fix an ordering on the monomials in $k[X_1, \ldots X_n]$. Then we can write an element f of $k[X_1, \ldots X_n]$ in a canonical fashion, by re-ordering its elements in decreasing order. For example, we would write

$$f = 4XY^2Z + 4Z^2 - 5X^3 + 7X^2Z^2$$

as

$$f = -5X^3 + 7X^2Z^2 + 4XY^2Z + 4Z^2 \quad (lex)$$

or

$$f = 4XY^2Z + 7X^2Z^2 - 5X^3 + 4Z^2 \quad (grevlex)$$

Let $\sum a_{\alpha} X^{\alpha} \in k[X_1, \ldots, X_n]$, in decreasing order:

$$f = a_{\alpha_0} X^{\alpha_0} +_{\alpha_1} X^{\alpha_1} + \ldots, \qquad \alpha_0 > \alpha_1 > \ldots, \quad \alpha_0 \neq 0$$

Then we define.

- The *multidegree* of f to be multdeg(f)= α_0 *;*
- The *leading coefficient of* f to be *LC(* f *)=* a_{α_0} *;*
- The *leading monomial of* f to be *LM(* f *)* = X^{α_0} *;*
- The *leading term of* f to be *LT(* f *)* = $a_{\alpha_0} X^{\alpha_0}$

For the polynomial $f = 4XY^2Z + ...,$ the multidegree is (1,2,1), the leading coefficient is 4, the leading monomial is XY^2Z, and the leading term is $4XY^2Z$. **The division algorithm in** $k[X_1,...X_n]$**.** Fix a monomial ordering in □ ^{2}. Suppose given a polynomial f and an ordered set $(g_1,...g_s)$ of polynomials; the division algorithm then constructs polynomials $a_1,...a_s$ and r such that $f = a_1g_1 + ... + a_sg_s + r$ Where either $r = 0$ or no monomial in r is divisible by any of $LT(g_1),...,LT(g_s)$ **Step 1:** If $LT(g_1) | LT(f)$, divide g_1 into f to get

$$f = a_1g_1 + h, \qquad a_1 = \frac{LT(f)}{LT(g_1)} \in k[X_1,...,X_n]$$

If $LT(g_1) | LT(h)$, repeat the process until $f = a_1g_1 + f_1$ (different a_1) with $LT(f_1)$ not divisible by $LT(g_1)$. Now divide g_2 into f_1, and so on, until $f = a_1g_1 + ... + a_sg_s + r_1$ With $LT(r_1)$ not divisible by any $LT(g_1),...LT(g_s)$ **Step 2:** Rewrite $r_1 = LT(r_1) + r_2$, and repeat Step 1 with r_2 for f : $f = a_1g_1 + ... + a_sg_s + LT(r_1) + r_3$ (different a_i's) **Monomial ideals.** In general, an ideal a will contain a polynomial without containing the individual terms of the polynomial; for example, the ideal $a = (Y^2 - X^3)$ contains $Y^2 - X^3$ but not Y^2 or X^3.

DEFINITION 1.5. An ideal a is *monomial* if $\sum c_\alpha X^\alpha \in a \Rightarrow X^\alpha \in a$ all α with $c_\alpha \neq 0$.

PROPOSITION 1.3. Let a be a *monomial ideal,* and let $A = \{\alpha \mid X^\alpha \in a\}$. Then A satisfies the condition $\alpha \in A, \ \beta \in$ □ $^n \Rightarrow \alpha + \beta \in$ $(*)$ And a is the k -subspace of $k[X_1,...,X_n]$ generated by the $X^\alpha, \alpha \in A$. Conversely, of A is a subset of □ ^{n} satisfying $(*)$, then the k-subspace a of $k[X_1,...,X_n]$ generated by $\{X^\alpha \mid \alpha \in A\}$ is a monomial ideal.

PROOF. It is clear from its definition that a monomial ideal a is the k -subspace of $k[X_1,...,X_n]$ generated by the set of monomials it contains. If $X^\alpha \in a$ and $X^\beta \in k[X_1,...,X_n]$.

If a permutation is chosen uniformly and at random from the $n!$ possible permutations in S_n, then the counts $C_j^{(n)}$ of cycles of length j are dependent random variables. The joint distribution of $C^{(n)} = (C_1^{(n)},...,C_n^{(n)})$ follows from Cauchy's formula, and is given by

$$P[C^{(n)} = c] = \frac{1}{n!}N(n,c) = 1\left\{\sum_{j=1}^{n} jc_j = n\right\}\prod_{j=1}^{n}(\frac{1}{j})^{c_j}\frac{1}{c_j!}, \qquad (1.1)$$

for $c \in$ □ $^n_+$.

Lemma1.7 For nonnegative integers $m_{1,...,}m_n$,

$$E\left(\prod_{j=1}^{n}(C_j^{(n)})^{[m_j]}\right) = \left(\prod_{j=1}^{n}\left(\frac{1}{j}\right)^{m_j}\right)1\left\{\sum_{j=1}^{n} jm_j \leq n\right\} \qquad (1.4)$$

Proof. This can be established directly by exploiting cancellation of the form $c_j^{[m_j]} / c_j^! = 1/(c_j - m_j)!$ when $c_j \geq m_j$, which occurs between the ingredients in Cauchy's formula and the falling factorials in the moments. Write $m = \sum jm_j$. Then, with the first sum indexed by $c = (c_1,...c_n) \in$ □ $^n_+$ and the last sum indexed by $d = (d_1,...,d_n) \in$ □ $^n_+$ via the correspondence $d_j = c_j - m_j$, we have

$$E\left(\prod_{j=1}^{n}(C_j^{(n)})^{[m_j]}\right) = \sum_c P[C^{(n)} = c]\prod_{j=1}^{n}(c_j)^{[m_j]}$$

$$= \sum_{c:c_j \geq m_j \ \ for\ all\ j} 1\left\{\sum_{j=1}^{n} jc_j = n\right\}\prod_{j=1}^{n}\frac{(c_j)^{[m_j]}}{j^{c_j}c_j!}$$

$$= \prod_{j=1}^{n}\frac{1}{j^{m_j}}\sum_d 1\left\{\sum_{j=1}^{n} jd_j = n - m\right\}\prod_{j=1}^{n}\frac{1}{j^{d_j}(d_j)!}$$

This last sum simplifies to the indicator $1(m \le n)$, corresponding to the fact that if $n-m \ge 0$, then $d_j = 0$ for $j > n-m$, and a random permutation in S_{n-m} must have some cycle structure $(d_1,\ldots,d_{n-m})$. The moments of $C_j^{(n)}$ follow immediately as

$$E(C_j^{(n)})^{[r]} = j^{-r} 1\{jr \le n\} \qquad (1.2)$$

We note for future reference that (1.4) can also be written in the form

$$E\left(\prod_{j=1}^{n} (C_j^{(n)})^{[m_j]}\right) = E\left(\prod_{j=1}^{n} Z_j^{[m_j]}\right) 1\left\{\sum_{j=1}^{n} jm_j \le n\right\}, \qquad (1.3)$$

Where the Z_j are independent Poisson-distribution random variables that satisfy $E(Z_j) = 1/j$

The marginal distribution of cycle counts provides a formula for the joint distribution of the cycle counts C_j^n, we find the distribution of C_j^n using a combinatorial approach combined with the inclusion-exclusion formula.

Lemma 1.8. For $1 \le j \le n$,

$$P[C_j^{(n)} = k] = \frac{j^{-k}}{k!} \sum_{l=0}^{[n/j]-k} (-1)^l \frac{j^{-l}}{l!} \qquad (1.1)$$

Proof. Consider the set I of all possible cycles of length j, formed with elements chosen from $\{1,2,\ldots n\}$, so that $|I| = n^{[j]/j}$. For each $\alpha \in I$, consider the "property" G_α of having α; that is, G_α is the set of permutations $\pi \in S_n$ such that α is one of the cycles of π. We then have $|G_\alpha| = (n-j)!$, since the elements of $\{1,2,\ldots,n\}$ not in α must be permuted among themselves. To use the inclusion-exclusion formula we need to calculate the term S_r, which is the sum of the probabilities of the r-fold intersection of properties, summing over all sets of r distinct properties. There are two cases to consider. If the r properties are indexed by r cycles having no elements in common, then the intersection specifies how rj elements are moved by the permutation, and there are $(n-rj)!1(rj \le n)$ permutations in the intersection. There are $n^{[rj]}/(j^r r!)$ such intersections. For the other case, some two distinct properties name some element in common, so no permutation can have both these properties, and the r-fold intersection is empty. Thus

$$S_r = (n-rj)!1(rj \le n)$$

$$\times \frac{n^{[rj]}}{j^r r!} \frac{1}{n!} = 1(rj \le n) \frac{1}{j^r r!}$$

Finally, the inclusion-exclusion series for the number of permutations having exactly k properties is

$$\sum_{l \ge 0} (-1)^l \binom{k+l}{l} S_{k+l},$$

Which simplifies to (1.1) Returning to the original hat-check problem, we substitute j=1 in (1.1) to obtain the distribution of the number of fixed points of a random permutation. For $k = 0,1,\ldots,n$,

$$P[C_1^{(n)} = k] = \frac{1}{k!} \sum_{l=0}^{n-k} (-1)^l \frac{1}{l!}, \qquad (1.2)$$

and the moments of $C_1^{(n)}$ follow from (1.2) with $j=1$. In particular, for $n \ge 2$, the mean and variance of $C_1^{(n)}$ are both equal to 1. The joint distribution of $(C_1^{(n)},\ldots,C_b^{(n)})$ for any $1 \le b \le n$ has an expression similar to (1.7); this too can be derived by inclusion-exclusion. For any $c = (c_1,\ldots,c_b) \in □_+^b$ with $m = \sum ic_i$,

$$P[(C_1^{(n)},\ldots,C_b^{(n)}) = c]$$

$$= \left\{\prod_{i=1}^{b} \left(\frac{1}{i}\right)^{c_i} \frac{1}{c_i!}\right\} \sum_{\substack{l \ge 0 \text{ with} \\ \sum il_i \le n-m}} (-1)^{l_1+\ldots+l_b} \prod_{i=1}^{b} \left(\frac{1}{i}\right)^{l_i} \frac{1}{l_i!} \qquad (1.3)$$

The joint moments of the first b counts $C_1^{(n)},\ldots,C_b^{(n)}$ can be obtained directly from (1.2) and (1.3) by setting $m_{b+1} = \ldots = m_n = 0$

The limit distribution of cycle counts

It follows immediately from Lemma 1.2 that for each fixed j, as $n \to \infty$,

$$P[C_j^{(n)} = k] \to \frac{j^{-k}}{k!} e^{-1/j}, \quad k = 0,1,2,\ldots,$$

So that $C_j^{(n)}$ converges in distribution to a random variable Z_j having a Poisson distribution with mean $1/j$; we use the notation $C_j^{(n)} \to_d Z_j$ where $Z_j □ P_o(1/j)$ to describe this. Infact, the limit random variables are independent.

Theorem 1.6 The process of cycle counts converges in distribution to a Poisson process of □ with intensity j^{-1}. That is, as $n \to \infty$,

$$(C_1^{(n)}, C_2^{(n)}, ...) \to_d (Z_1, Z_2, ...) \qquad (1.1)$$

Where the $Z_j, j = 1, 2, ...,$ are independent Poisson-distributed random variables with

$$E(Z_j) = \frac{1}{j}$$

Proof. To establish the converges in distribution one shows that for each fixed $b \geq 1,$ as $n \to \infty,$

$$P[(C_1^{(n)}, ..., C_b^{(n)}) = c] \to P[(Z_1, ..., Z_b) = c]$$

Error rates

The proof of Theorem says nothing about the rate of convergence. Elementary analysis can be used to estimate this rate when $b = 1$. Using properties of alternating series with decreasing terms, for $k = 0, 1, ..., n,$

$$\frac{1}{k!}\left(\frac{1}{(n-k+1)!} - \frac{1}{(n-k+2)!}\right) \leq \left|P[C_1^{(n)} = k] - P[Z_1 = k]\right|$$

$$\leq \frac{1}{k!(n-k+1)!}$$

It follows that

$$\frac{2^{n+1}}{(n+1)!}\frac{n}{n+2} \leq \sum_{k=0}^{n}\left|P[C_1^{(n)} = k] - P[Z_1 = k]\right| \leq \frac{2^{n+1}-1}{(n+1)!} \qquad (1.11)$$

Since

$$P[Z_1 > n] = \frac{e^{-1}}{(n+1)!}\left(1 + \frac{1}{n+2} + \frac{1}{(n+2)(n+3)} + ...\right) < \frac{1}{(n+1)!},$$

We see from (1.11) that the total variation distance between the distribution $L(C_1^{(n)})$ of $C_1^{(n)}$ and the distribution $L(Z_1)$ of Z_1

Establish the asymptotics of $\mathrm{P}\left[A_n(C^{(n)})\right]$ under conditions (A_0) and $(B_{01}),$ where

$$A_n(C^{(n)}) = \bigcap_{1 \leq i \leq n} \bigcap_{r_i^{'}+1 \leq j \leq r_i} \left\{C_{ij}^{(n)} = 0\right\},$$

and $\zeta_i = (r_i^{'} / r_{id}) - 1 = O(i^{-g^{'}})$ as $i \to \infty,$ for some $g^{'} > 0.$ We start with the expression

$$P[A_n(C^{(n)})] = \frac{P[T_{0m}(Z^{'}) = n]}{P[T_{0m}(Z) = n]}$$

$$\prod_{\substack{1 \leq i \leq n \\ r_i^{'}+1 \leq j \leq r_i}} \left\{1 - \frac{\theta}{ir_i}(1 + E_{i0})\right\} \qquad (1.1)$$

$$P[T_{0n}(Z^{'}) = n]$$

$$= \frac{\theta d}{n} \exp\left\{\sum_{i \geq 1}[\log(1 + i^{-1}\theta d) - i^{-1}\theta d]\right\}$$

$$\left\{1 + O(n^{-1}\varphi_{\{1,2,7\}}^{'}(n))\right\} \qquad (1.2)$$

and

$$P[T_{0n}(Z^{'}) = n]$$

$$= \frac{\theta d}{n} \exp\left\{\sum_{i \geq 1}[\log(1 + i^{-1}\theta d) - i^{-1}\theta d]\right\}$$

$$\left\{1 + O(n^{-1}\varphi_{\{1,2,7\}}(n))\right\} \qquad (1.3)$$

Where $\varphi_{\{1,2,7\}}^{'}(n)$ refers to the quantity derived from $Z^{'}$. It thus follows that $P[A_n(C^{(n)})] □ Kn^{-\theta(1-d)}$ for a constant K, depending on Z and the $r_i^{'}$ and computable explicitly from (1.1) – (1.3), if Conditions (A_0) and (B_{01}) are satisfied and if $\zeta_i^{*} = O(i^{-g^{'}})$ from some $g^{'} > 0,$ since, under these circumstances, both $n^{-1}\varphi_{\{1,2,7\}}^{'}(n)$ and $n^{-1}\varphi_{\{1,2,7\}}(n)$ tend to zero as $n \to \infty.$ In particular, for polynomials and square free polynomials, the relative error in this asymptotic approximation is of order n^{-1} if $g^{'} > 1.$

For $0 \leq b \leq n/8$ and $n \geq n_0,$ with n_0

$$d_{TV}(L(C[1,b]), L(Z[1,b]))$$

$$\leq d_{TV}(L(\overset{□}{C}[1,b]), L(\overset{□}{Z}[1,b]))$$

$$\leq \varepsilon_{\{7,7\}}(n,b),$$

Where $\varepsilon_{\{7,7\}}(n,b) = O(b/n)$ under Conditions $(A_0), (D_1)$ and (B_{11}) Since, by the Conditioning Relation,

$$L(\overset{□}{C}[1,b] \mid T_{0b}(C) = l) = L(\overset{□}{Z}[1,b] \mid T_{0b}(Z) = l),$$

It follows by direct calculation that

$$d_{TV}(L(\overset{□}{C}[1,b]), L(\overset{□}{Z}[1,b]))$$

$$= d_{TV}(L(T_{0b}(C)), L(T_{0b}(Z)))$$

$$= \max_{A} \sum_{r \in A} P[T_{0b}(Z) = r]$$

$$\left\{1 - \frac{P[T_{bn}(Z) = n - r]}{P[T_{0n}(Z) = n]}\right\} \qquad (1.4)$$

Suppressing the argument Z from now on, we thus obtain

$$d_{TV}(L(\tilde{C}[1,b]), L(\tilde{Z}[1,b]))$$
$$= \sum_{r\ge 0} P[T_{0b}=r]\left\{1-\frac{P[T_{bn}=n-r]}{P[T_{0n}=n]}\right\}_+$$
$$\le \sum_{r>n/2} P[T_{0b}=r] + \sum_{r=0}^{[n/2]} \frac{P[T_{0b}=r]}{P[T_{0b}=n]}$$
$$\times \left\{\sum_{s=0}^{n} P[T_{0b}=s](P[T_{bn}=n-s]-P[T_{bn}=n-r]\right\}_+$$

$$\le \sum_{r>n/2} P[T_{0b}=r] + \sum_{r=0}^{[n/2]} P[T_{0b}=r]$$
$$\times \sum_{s=0}^{[n/2]} P[T_{0b}=s]\frac{\{P[T_{bn}=n-s]-P[T_{bn}=n-r]\}}{P[T_{0n}=n]}$$
$$+\sum_{s=0}^{[n/2]} P[T_{0b}=r]\sum_{s=[n/2]+1}^{n} P[T=s]P[T_{bn}=n-s]/P[T_{0n}=n]$$

The first sum is at most $2n^{-1}ET_{0b}$; the third is bound by

$$(\max_{n/2<s\le n} P[T_{0b}=s])/P[T_{0n}=n]$$
$$\le \frac{2\varepsilon_{\{10.5(1)\}}(n/2,b)}{n}\frac{3n}{\theta P_\theta[0,1]},$$
$$\frac{3n}{\theta P_\theta[0,1]}4n^{-2}\phi^*_{\{10.8\}}(n)\sum_{r=0}^{[n/2]}P[T_{0b}=r]\sum_{s=0}^{[n/2]}P[T_{0b}=s]\frac{1}{2}|r-s|$$
$$\le \frac{12\phi^*_{\{10.8\}}(n)}{\theta P_\theta[0,1]}\frac{ET_{0b}}{n}$$

Hence we may take

$$\varepsilon_{\{7,7\}}(n,b) = 2n^{-1}ET_{0b}(Z)\left\{1+\frac{6\phi^*_{\{10.8\}}(n)}{\theta P_\theta[0,1]}\right\}P$$
$$+\frac{6}{\theta P_\theta[0,1]}\varepsilon_{\{10.5(1)\}}(n/2,b) \qquad (1.5)$$

Required order under Conditions $(A_0),(D_1)$ and (B_{11}), if $S(\infty)<\infty$. If not, $\phi^*_{\{10.8\}}(n)$ can be replaced by $\phi^*_{\{10.11\}}(n)$ in the above, which has the required order, without the restriction on the r_i implied by $S(\infty)<\infty$. Examining the Conditions $(A_0),(D_1)$ and (B_{11}), it is perhaps surprising to find that (B_{11}) is required instead of just (B_{01}); that is, that we should need $\sum_{l\ge 2} l\varepsilon_{il} = O(i^{-a_1})$ to hold for some $a_1>1$. A first observation is that a similar problem arises with the rate of decay of ε_{i1} as well. For this reason, n_1 is replaced by $\tilde{n}_1$. This makes it possible to replace condition (A_1) by the weaker pair of conditions (A_0) and (D_1) in the eventual assumptions needed for $\varepsilon_{\{7,7\}}(n,b)$ to be of order $O(b/n)$; the decay rate requirement of order $i^{-1-\gamma}$ is shifted from ε_{i1} itself to its first difference. This is needed to obtain the right approximation error for the random mappings example. However, since all the classical applications make far more stringent assumptions about the $\varepsilon_{i1}, l\ge 2,$ than are made in (B_{11}). The critical point of the proof is seen where the initial estimate of the difference $P[T^{(m)}_{bn}=s]-P[T^{(m)}_{bn}=s+1]$. The factor $\varepsilon_{\{10.10\}}(n),$ which should be small, contains a far tail element from $\tilde{n}_1$ of the form $\phi^\theta_1(n)+u^*_1(n),$ which is only small if $a_1>1,$ being otherwise of order $O(n^{1-a_1+\delta})$ for any $\delta>0,$ since $a_2>1$ is in any case assumed. For $s\ge n/2,$ this gives rise to a contribution of order $O(n^{-1-a_1+\delta})$ in the estimate of the difference $P[T_{bn}=s]-P[T_{bn}=s+1],$ which, in the remainder of the proof, is translated into a contribution of order $O(tn^{-1-a_1+\delta})$ for differences of the form $P[T_{bn}=s]-P[T_{bn}=s+1],$ finally leading to a contribution of order $bn^{-a_1+\delta}$ for any $\delta>0$ in $\varepsilon_{\{7.7\}}(n,b).$ Some improvement would seem to be possible, defining the function g by $g(w)=1_{\{w=s\}}-1_{\{w=s+t\}},$ differences that are of the form $P[T_{bn}=s]-P[T_{bn}=s+t]$ can be directly estimated, at a cost of only a single contribution of the form $\phi^\theta_1(n)+u^*_1(n).$ Then, iterating the cycle, in which one estimate of a difference in point probabilities is improved to an estimate of smaller order, a bound of the form

$$|P[T_{bn}=s]-P[T_{bn}=s+t]| = O(n^{-2}t+n^{-1-a_1+\delta})$$

for any $\delta>0$ could perhaps be attained, leading to a final error estimate in order $O(bn^{-1}+n^{-a_1+\delta})$ for any $\delta>0$, to replace $\varepsilon_{\{7.7\}}(n,b).$ This would be of the ideal order $O(b/n)$ for large enough $b,$ but would still be coarser for small $b.$

With b and n as in the previous section, we wish to show that

$$\left| d_{TV}(L(C[1,b]),L(Z[1,b])) - \frac{1}{2}(n+1)^{-1}|1-\theta| E\left|T_{0b} - ET_{0b}\right| \right|$$

$$\leq \varepsilon_{\{7,8\}}(n,b),$$

Where $\varepsilon_{\{7.8\}}(n,b) = O(n^{-1}b[n^{-1}b + n^{-\beta_{12}+\delta}])$ for any $\delta > 0$ under Conditions $(A_0),(D_1)$ and (B_{12}), with β_{12}. The proof uses sharper estimates. As before, we begin with the formula

$$d_{TV}(L(\overset{\square}{C}[1,b]),L(\overset{\square}{Z}[1,b]))$$

$$= \sum_{r\geq 0} P[T_{0b}=r]\left\{1 - \frac{P[T_{bn}=n-r]}{P[T_{0n}=n]}\right\}_{+}$$

Now we observe that

$$\left| \sum_{r\geq 0} P[T_{0b}=r]\left\{1 - \frac{P[T_{bn}=n-r]}{P[T_{0n}=n]}\right\}_{+} - \sum_{r=0}^{[n/2]} \frac{P[T_{0b}=r]}{P[T_{0n}=n]} \right|$$

$$\times \left| \sum_{s=[n/2]+1}^{n} P[T_{0b}=s](P[T_{bn}=n-s] - P[T_{bn}=n-r]) \right|$$

$$\leq 4n^{-2}ET_{0b}^{2} + (\max_{n/2<s\leq n} P[T_{0b}=s]) / P[T_{0n}=n]$$

$$+P[T_{0b} > n/2]$$

$$\leq 8n^{-2}ET_{0b}^{2} + \frac{3\varepsilon_{\{10.5(2)\}}(n/2,b)}{\theta P_{\theta}[0,1]}, \qquad (1.1)$$

We have

$$\left| \sum_{r=0}^{[n/2]} \frac{P[T_{0b}=r]}{P[T_{0n}=n]} \right.$$

$$\times \left(\left\{ \sum_{s=0}^{[n/2]} P[T_{0b}=s](P[T_{bn}=n-s] - P[T_{bn}=n-r] \right\}_{+} \right.$$

$$\left. - \left\{ \sum_{s=0}^{[n/2]} P[T_{0b}=s] \frac{(s-r)(1-\theta)}{n+1} P[T_{0n}=n] \right\}_{+} \right) \right|$$

$$\leq \frac{1}{n^{2}P[T_{0n}=n]} \sum_{r\geq 0} P[T_{0b}=r] \sum_{s\geq 0} P[T_{0b}=s]|s-r|$$

$$\times \left\{ \varepsilon_{\{10.14\}}(n,b) + 2(r\vee s)|1-\theta| n^{-1}\left\{K_{0}\theta + 4\phi^{*}_{\{10.8\}}(n)\right\}\right\}$$

$$\leq \frac{6}{\theta n P_{\theta}[0,1]} ET_{0b}\varepsilon_{\{10.14\}}(n,b)$$

$$+4|1-\theta| n^{-2}ET_{0b}^{2}\left\{K_{0}\theta + 4\phi^{*}_{\{10.8\}}(n)\right\}$$

$$\left(\frac{3}{\theta n P_{\theta}[0,1]}\right) \Big\}, \qquad (1.2)$$

The approximation in (1.2) is further simplified by noting that

$$\sum_{r=0}^{[n/2]} P[T_{0b}=r] \left| \left\{ \sum_{s=0}^{[n/2]} P[T_{0b}=s] \frac{(s-r)(1-\theta)}{n+1} \right\}_{+} \right.$$

$$\left. - \left\{ \sum_{s=0} P[T_{0b}=s] \frac{(s-r)(1-\theta)}{n+1} \right\}_{+} \right|$$

$$\leq \sum_{r=0}^{[n/2]} P[T_{0b}=r] \sum_{s>[n/2]} P[T_{0b}=s] \frac{(s-r)|1-\theta|}{n+1}$$

$$\leq |1-\theta| n^{-1}E(T_{0b}1\{T_{0b} > n/2\}) \leq 2|1-\theta| n^{-2}ET_{0b}^{2}, \qquad (1.3)$$

and then by observing that

$$\sum_{r>[n/2]} P[T_{0b}=r] \left\{ \sum_{s\geq 0} P[T_{0b}=s] \frac{(s-r)(1-\theta)}{n+1} \right\}$$

$$\leq n^{-1}|1-\theta| (ET_{0b}P[T_{0b} > n/2] + E(T_{0b}1\{T_{0b} > n/2\}))$$

$$\leq 4|1-\theta| n^{-2}ET_{0b}^{2} \qquad (1.4)$$

Combining the contributions of (1.2) –(1.3), we thus find tha

$$\begin{aligned}&\Big|\, d_{TV}(L(\overset{\square}{C}[1,b]), L(\overset{\square}{Z}[1,b]))\\ &-(n+1)^{-1}\sum_{r\ge0} P[T_{0b}=r]\left\{\sum_{s\ge0}P[T_{0b}=s](s-r)(1-\theta)\right\}_+ \Big|\\ &\le \varepsilon_{\{7.8\}}(n,b)\\ &=\frac{3}{\theta P_\theta[0,1]}\left\{\varepsilon_{\{10.5(2)\}}(n/2,b)+2n^{-1}ET_{0b}\varepsilon_{\{10.14\}}(n,b)\right\}\\ &+2n^{-2}ET_{0b}^2\left\{4+3|1-\theta|+\frac{24|1-\theta|\phi^*_{\{10.8\}}(n)}{\theta P_\theta[0,1]}\right\}\end{aligned} \tag{1.5}$$

The quantity $\varepsilon_{\{7.8\}}(n,b)$ is seen to be of the order claimed under Conditions $(A_0),(D_1)$ and (B_{12}) , provided that $S(\infty)<\infty;$ this supplementary condition can be removed if $\phi^*_{\{10.8\}}(n)$ is replaced by $\phi^*_{\{10.11\}}(n)$ in the definition of $\varepsilon_{\{7.8\}}(n,b)$, has the required order without the restriction on the r_i implied by assuming that $S(\infty)<\infty$. Finally, a direct calculation now shows that

$$\sum_{r\ge0}P[T_{0b}=r]\left\{\sum_{s\ge0}P[T_{0b}=s](s-r)(1-\theta)\right\}_+ =\frac{1}{2}|1-\theta|E|T_{0b}-ET_{0b}|$$

Example 1.0. Consider the point $O=(0,...,0)\in$ □ ^{n}. For an arbitrary vector r , the coordinates of the point $x=O+r$ are equal to the respective coordinates of the vector $r: x=(x^1,...x^n)$ and $r=(x^1,...,x^n)$. The vector r such as in the example is called the position vector or the radius vector of the point x . (Or, in greater detail: r is the radius-vector of x w.r.t an origin O). Points are frequently specified by their radius-vectors. This presupposes the choice of O as the “standard origin”. Let us summarize. We have considered □ ^{n} and interpreted its elements in two ways: as points and as vectors. Hence we may say that we leading with the two copies of □ ^{n}: □ ^{n} = {points}, □ ^{n} = {vectors}

Operations with vectors: multiplication by a number, addition. Operations with points and vectors: adding a vector to a point (giving a point), subtracting two points (giving a vector). □ ^{n} treated in this way is called an *n-dimensional affine space*. *(*An “abstract” affine space is a pair of sets , the set of points and the set of vectors so that the operations as above are defined axiomatically). Notice that vectors in an affine space are also known as “free vectors”. Intuitively, they are not fixed at points and “float freely” in space. From □ ^{n} considered as an affine space we can precede in two opposite directions: □ ^{n} as an Euclidean space $\Leftarrow$ □ ^{n} as an affine space $\Rightarrow$ □ ^{n} as a manifold.Going to the left means introducing some extra structure which will make the geometry richer. Going to the right means forgetting about part of the affine structure; going further in this direction will lead us to the so-called “smooth (or differentiable) manifolds”. The theory of differential forms does not require any extra geometry. So our natural direction is to the right. The Euclidean structure, however, is useful for examples and applications. So let us say a few words about it:

Remark 1.0. *Euclidean geometry.* In □ ^{n} considered as an affine space we can already do a good deal of geometry. For example, we can consider lines and planes, and quadric surfaces like an ellipsoid. However, we cannot discuss such things as “lengths”, “angles” or “areas” and “volumes”. To be able to do so, we have to introduce some more definitions, making □ ^{n} a Euclidean space. Namely, we define the length of a vector $a=(a^1,...,a^n)$ to be

$$|a|:=\sqrt{(a^1)^2+...+(a^n)^2} \tag{1}$$

After that we can also define distances between points as follows:

$$d(A,B):=|\overrightarrow{AB}| \tag{2}$$

One can check that the distance so defined possesses natural properties that we expect: is it always non-negative and equals zero only for coinciding points; the distance from A to B is the same as that from B to A (symmetry); also, for three points, A, B and C, we have $d(A,B)\le d(A,C)+d(C,B)$ (the “triangle inequality”). To define angles, we first introduce the scalar product of two vectors

$$(a,b):=a^1b^1+...+a^nb^n \tag{3}$$

Thus $|a|=\sqrt{(a,a)}$. The scalar product is also denote by dot: $a.b=(a,b)$, and hence is often referred to as the “dot product” . Now, for nonzero vectors, we define the angle between them by the equality

$$\cos\alpha:=\frac{(a,b)}{|a||b|} \tag{4}$$

The angle itself is defined up to an integral multiple of 2π . For this definition to be consistent we have to ensure that the r.h.s. of (4) does not exceed 1 by the absolute value. This follows from the inequality

$$(a,b)^2 \le |a|^2 |b|^2 \qquad (5)$$

known as the Cauchy–Bunyakovsky–Schwarz inequality (various combinations of these three names are applied in different books). One of the ways of proving (5) is to consider the scalar square of the linear combination $a+tb$, where $t \in R$. As $(a+tb, a+tb) \ge 0$ is a quadratic polynomial in t which is never negative, its discriminant must be less or equal zero. Writing this explicitly yields (5). The triangle inequality for distances also follows from the inequality (5).

Example 1.1. Consider the function $f(x) = x^i$ (the i-th coordinate). The linear function dx^i (the differential of x^i) applied to an arbitrary vector h is simply h^i.From these examples follows that we can rewrite df as

$$df = \frac{\partial f}{\partial x^1} dx^1 + \ldots + \frac{\partial f}{\partial x^n} dx^n, \qquad (1)$$

which is the standard form. Once again: the partial derivatives in (1) are just the coefficients (depending on x); $dx^1, dx^2, \ldots$ are linear functions giving on an arbitrary vector h its coordinates $h^1, h^2, \ldots,$ respectively. Hence

$$df(x)(h) = \partial_{hf(x)} = \frac{\partial f}{\partial x^1} h^1 + \ldots + \frac{\partial f}{\partial x^n} h^n, \qquad (2)$$

Theorem 1.7. Suppose we have a parametrized curve $t \mapsto x(t)$ passing through $x_0 \in \mathbb{R}^n$ at $t = t_0$ and with the velocity vector $x(t_0) = \upsilon$ Then

$$\frac{df(x(t))}{dt}(t_0) = \partial_\upsilon f(x_0) = df(x_0)(\upsilon) \qquad (1)$$

Proof. Indeed, consider a small increment of the parameter $t : t_0 \mapsto t_0 + \Delta t$, Where $\Delta t \mapsto 0$. On the other hand, we have $f(x_0+h) - f(x_0) = df(x_0)(h) + \beta(h)|h|$ for an arbitrary vector h, where $\beta(h) \to 0$ when $h \to 0$. Combining it together, for the increment of $f(x(t))$ we obtain

$$\begin{aligned} & f(x(t_0+\Delta t) - f(x_0) \\ & = df(x_0)(\upsilon.\Delta t + \alpha(\Delta t)\Delta t) \\ & + \beta(\upsilon.\Delta t + \alpha(\Delta t)\Delta t).|\upsilon\Delta t + \alpha(\Delta t)\Delta t| \\ & = df(x_0)(\upsilon).\Delta t + \gamma(\Delta t)\Delta t \end{aligned}$$

For a certain $\gamma(\Delta t)$ such that $\gamma(\Delta t) \to 0$ when $\Delta t \to 0$ (we used the linearity of $df(x_0)$). By the definition, this means that the derivative of $f(x(t))$ at $t = t_0$ is exactly $df(x_0)(\upsilon)$. The statement of the theorem can be expressed by a simple formula:

$$\frac{df(x(t))}{dt} = \frac{\partial f}{\partial x^1} x^1 + \ldots + \frac{\partial f}{\partial x^n} x^n \qquad (2)$$

To calculate the value Of df at a point x_0 on a given vector υ one can take an arbitrary curve passing Through x_0 at t_0 with υ as the velocity vector at t_0 and calculate the usual derivative of $f(x(t))$ at $t = t_0$.

Theorem 1.8. For functions $f, g : U \to \mathbb{R}$, $U \subset \mathbb{R}^n$,

$$d(f+g) = df + dg \qquad (1)$$

$$d(fg) = df.g + f.dg \qquad (2)$$

Proof. Consider an arbitrary point x_0 and an arbitrary vector υ stretching from it. Let a curve $x(t)$ be such that $x(t_0) = x_0$ and $x(t_0) = \upsilon$.
Hence

$$d(f+g)(x_0)(\upsilon) = \frac{d}{dt}(f(x(t)) + g(x(t)))$$

at $t = t_0$ and

$$d(fg)(x_0)(\upsilon) = \frac{d}{dt}(f(x(t))g(x(t)))$$

at $t = t_0$ Formulae (1) and (2) then immediately follow from the corresponding formulae for the usual derivative Now, almost without change the theory generalizes to functions taking values in $\mathbb{R}^m$ instead of $\mathbb{R}$. The only difference is that now the differential of a map $F : U \to \mathbb{R}^m$ at a point x will be a linear function taking vectors in $\mathbb{R}^n$ to vectors in $\mathbb{R}^m$ (instead of $\mathbb{R}$) . For an arbitrary vector $h \in| \mathbb{R}^n$,

$$F(x+h)=F(x)+dF(x)(h)$$
$$+\beta(h)|h| \qquad (3)$$

Where $\beta(h)\to 0$ when $h\to 0$. We have $dF=(dF^1,...,dF^m)$ and

$$dF=\frac{\partial F}{\partial x^1}dx^1+...+\frac{\partial F}{\partial x^n}dx^n$$
$$=\begin{pmatrix}\frac{\partial F^1}{\partial x^1}\cdots\cdot\frac{\partial F^1}{\partial x^n}\\ \ldots\ \ldots\ \ldots\\ \frac{\partial F^m}{\partial x^1}\cdots\frac{\partial F^m}{\partial x^n}\end{pmatrix}\begin{pmatrix}dx^1\\ \ldots\\ dx^n\end{pmatrix} \qquad (4)$$

In this matrix notation we have to write vectors as vector-columns.

Theorem 1.9. For an arbitrary parametrized curve $x(t)$ in $\square^n$, the differential of a map $F:U\to\square^m$ (where $U\subset\square^n$) maps the velocity vector $x(t)$ to the velocity vector of the curve $F(x(t))$ in $\square^m$:

$$\frac{dF(x(t))}{dt}=dF(x(t))(\dot{x}(t)) \qquad (1)$$

Proof. By the definition of the velocity vector,

$$x(t+\Delta t)=x(t)+\dot{x}(t).\Delta t+\alpha(\Delta t)\Delta t \qquad (2)$$

Where $\alpha(\Delta t)\to 0$ when $\Delta t\to 0$. By the definition of the differential,

$$F(x+h)=F(x)+dF(x)(h)+\beta(h)|h \qquad (3)|$$

Where $\beta(h)\to 0$ when $h\to 0$. we obtain

$$F(x(t+\Delta t))=F(x+\underbrace{\dot{x}(t).\Delta t+\alpha(\Delta t)\Delta t)}_{h}$$

$$=F(x)+dF(x)(\dot{x}(t)\Delta t+\alpha(\Delta t)\Delta t)+$$

$$\beta(\dot{x}(t)\Delta t+\alpha(\Delta t)\Delta t).\left|\dot{x}(t)\Delta t+\alpha(\Delta t)\Delta t\right|$$

$$=F(x)+dF(x)(\dot{x}(t)\Delta t+\gamma(\Delta t)\Delta t$$

For some $\gamma(\Delta t)\to 0$ when $\Delta t\to 0$. This precisely means that $dF(x)\dot{x}(t)$ is the velocity vector of $F(x)$. As every vector attached to a point can be viewed as the velocity vector of some curve passing through this point, this theorem gives a clear geometric picture of dF as a linear map on vectors.

Theorem 1.10 Suppose we have two maps $F:U\to V$ and $G:V\to W,$ where $U\subset\square^n, V\subset\square^m, W\subset\square^p$ (open domains). Let $F:x\mapsto y=F(x)$. Then the differential of the composite map $GoF:U\to W$ is the composition of the differentials of F and G:

$$d(GoF)(x)=dG(y)odF(x) \qquad (4)$$

Proof. We can use the description of the differential .Consider a curve $x(t)$ in $\square^n$ with the velocity vector $\dot{x}$. Basically, we need to know to which vector in $\square^p$ it is taken by $d(GoF)$. the curve $(GoF)(x(t)=G(F(x(t))$. By the same theorem, it equals the image under dG of the Anycast Flow vector to the curve $F(x(t))$ in $\square^m$. Applying the theorem once again, we see that the velocity vector to the curve $F(x(t))$ is the image under dF of the vector $\dot{x}(t)$. Hence $d(GoF)(\dot{x})=dG(dF(\dot{x}))$ for an arbitrary vector $\dot{x}$.

Corollary 1.0. If we denote coordinates in $\square^n$ by $(x^1,...,x^n)$ and in $\square^m$ by $(y^1,...,y^m)$, and write

$$dF=\frac{\partial F}{\partial x^1}dx^1+...+\frac{\partial F}{\partial x^n}dx^n \qquad (1)$$

$$dG=\frac{\partial G}{\partial y^1}dy^1+...+\frac{\partial G}{\partial y^n}dy^n, \qquad (2)$$

Then the chain rule can be expressed as follows:

$$d(GoF)=\frac{\partial G}{\partial y^1}dF^1+...+\frac{\partial G}{\partial y^m}dF^m, \qquad (3)$$

Where dF^i are taken from (1). In other words, to get $d(GoF)$ we have to substitute into (2) the expression for $dy^i=dF^i$ from (3). This can also be expressed by the following matrix formula:

$$d(GoF)=\begin{pmatrix}\frac{\partial G^1}{\partial y^1}\cdots\cdot\frac{\partial G^1}{\partial y^m}\\ \ldots\ \ldots\ \ldots\\ \frac{\partial G^p}{\partial y^1}\cdots\frac{\partial G^p}{\partial y^m}\end{pmatrix}\begin{pmatrix}\frac{\partial F^1}{\partial x^1}\cdots\cdot\frac{\partial F^1}{\partial x^n}\\ \ldots\ \ldots\ \ldots\\ \frac{\partial F^m}{\partial x^1}\cdots\frac{\partial F^m}{\partial x^n}\end{pmatrix}\begin{pmatrix}dx^1\\ \ldots\\ dx^n\end{pmatrix} \qquad (4)$$

i.e., if dG and dF are expressed by matrices of partial derivatives, then $d(GoF)$ is expressed by the product of these matrices. This is often written as

$$\begin{pmatrix} \frac{\partial z^1}{\partial x^1} \cdots \frac{\partial z^1}{\partial x^n} \\ \cdots \quad \cdots \quad \cdots \\ \frac{\partial z^p}{\partial x^1} \cdots \frac{\partial z^p}{\partial x^n} \end{pmatrix} = \begin{pmatrix} \frac{\partial z^1}{\partial y^1} \cdots \frac{\partial z^1}{\partial y^m} \\ \cdots \quad \cdots \quad \cdots \\ \frac{\partial z^p}{\partial y^1} \cdots \frac{\partial z^p}{\partial y^m} \end{pmatrix} \begin{pmatrix} \frac{\partial y^1}{\partial x^1} \cdots \frac{\partial y^1}{\partial x^n} \\ \cdots \quad \cdots \quad \cdots \\ \frac{\partial y^m}{\partial x^1} \cdots \frac{\partial y^m}{\partial x^n} \end{pmatrix}, \qquad (5)$$

Or

$$\frac{\partial z^\mu}{\partial x^a} = \sum_{i=1}^{m} \frac{\partial z^\mu}{\partial y^i}\frac{\partial y^i}{\partial x^a}, \qquad (6)$$

Where it is assumed that the dependence of $y \in \mathbb{R}^m$ on $x \in \mathbb{R}^n$ is given by the map F, the dependence of $z \in \mathbb{R}^p$ on $y \in \mathbb{R}^m$ is given by the map G, and the dependence of $z \in \mathbb{R}^p$ on $x \in \mathbb{R}^n$ is given by the composition GoF.

Definition 1.6. Consider an open domain $U \subset \mathbb{R}^n$. Consider also another copy of $\mathbb{R}^n$, denoted for distinction $\mathbb{R}^n_y$, with the standard coordinates $(y^1 ... y^n)$. A system of coordinates in the open domain U is given by a map $F: V \to U$, where $V \subset \mathbb{R}^n_y$ is an open domain of $\mathbb{R}^n_y$, such that the following three conditions are satisfied :

(1) F is smooth;

(2) F is invertible;

(3) $F^{-1}: U \to V$ is also smooth

The coordinates of a point $x \in U$ in this system are the standard coordinates of $F^{-1}(x) \in \mathbb{R}^n_y$

In other words,

$$F: (y^1 ..., y^n) \mapsto x = x(y^1 ..., y^n) \qquad (1)$$

Here the variables $(y^1 ..., y^n)$ are the "new" coordinates of the point x

Example 1.2. Consider a curve in $\mathbb{R}^2$ specified in polar coordinates as

$$x(t): r = r(t), \varphi = \varphi(t) \qquad (1)$$

We can simply use the chain rule. The map $t \mapsto x(t)$ can be considered as the composition of the maps $t \mapsto (r(t), \varphi(t)), (r, \varphi) \mapsto x(r, \varphi)$. Then, by the chain rule, we have

$$\dot{x} = \frac{dx}{dt} = \frac{\partial x}{\partial r}\frac{dr}{dt} + \frac{\partial x}{\partial \varphi}\frac{d\varphi}{dt} = \frac{\partial x}{\partial r}\dot{r} + \frac{\partial x}{\partial \varphi}\dot{\varphi} \qquad (2)$$

Here $\dot{r}$ and $\dot{\varphi}$ are scalar coefficients depending on t, whence the partial derivatives $\partial x / \partial r, \partial x / \partial \varphi$ are vectors depending on point in $\mathbb{R}^2$. We can compare this with the formula in the "standard" coordinates: $\dot{x} = e_1 \dot{x} + e_2 \dot{y}$. Consider the vectors $\partial x / \partial r, \partial x / \partial \varphi$. Explicitly we have

$$\frac{\partial x}{\partial r} = (\cos\varphi, \sin\varphi) \qquad (3)$$

$$\frac{\partial x}{\partial \varphi} = (-r\sin\varphi, r\cos\varphi) \qquad (4)$$

From where it follows that these vectors make a basis at all points except for the origin (where $r = 0$). It is instructive to sketch a picture, drawing vectors corresponding to a point as starting from that point. Notice that $\partial x / \partial r, \partial x / \partial \varphi$ are, respectively, the velocity vectors for the curves $r \mapsto x(r, \varphi)$ $(\varphi = \varphi_0\ fixed)$ and $\varphi \mapsto x(r, \varphi)$ $(r = r_0\ fixed)$. We can conclude that for an arbitrary curve given in polar coordinates the velocity vector will have components $(\dot{r}, \dot{\varphi})$ if as a basis we take $e_r := \partial x / \partial r, e_\varphi := \partial x / \partial \varphi$:

$$\dot{x} = e_r \dot{r} + e_\varphi \dot{\varphi} \qquad (5)$$

A characteristic feature of the basis e_r, e_φ is that it is not "constant" but depends on point. Vectors "stuck to points" when we consider curvilinear coordinates.

Proposition 1.3. The velocity vector has the same appearance in all coordinate systems.

Proof. Follows directly from the chain rule and the transformation law for the basis e_i.In particular, the elements of the basis $e_i = \partial x / \partial x^i$ (originally, a formal notation) can be understood directly as the velocity vectors of the coordinate lines

$x^i \mapsto x(x^1,...,x^n)$ (all coordinates but x^i are fixed). Since we now know how to handle velocities in arbitrary coordinates, the best way to treat the differential of a map $F:\square^n \to \square^m$ is by its action on the velocity vectors. By definition, we set

$$dF(x_0):\frac{dx(t)}{dt}(t_0)\mapsto\frac{dF(x(t))}{dt}(t_0) \qquad (1)$$

Now $dF(x_0)$ is a linear map that takes vectors attached to a point $x_0\in\square^n$ to vectors attached to the point $F(x)\in\square^m$

$$dF=\frac{\partial F}{\partial x^1}dx^1+...+\frac{\partial F}{\partial x^n}dx^n$$

$$(e_1,...,e_m)\begin{pmatrix} \frac{\partial F^1}{\partial x^1}\cdots\frac{\partial F^1}{\partial x^n}\\ ...\ \ ...\ \ ...\\ \frac{\partial F^m}{\partial x^1}\cdots\frac{\partial F^m}{\partial x^n}\end{pmatrix}\begin{pmatrix}dx^1\\ ...\\ dx^n\end{pmatrix}, \qquad (2)$$

In particular, for the differential of a function we always have

$$df=\frac{\partial f}{\partial x^1}dx^1+...+\frac{\partial f}{\partial x^n}dx^n, \qquad (3)$$

Where x^i are arbitrary coordinates. The form of the differential does not change when we perform a change of coordinates.

Example 1.3 Consider a 1-form in $\square^2$ given in the standard coordinates:

$A=-ydx+xdy$ In the polar coordinates we will have $x=r\cos\varphi, y=r\sin\varphi$, hence

$$dx=\cos\varphi dr-r\sin\varphi d\varphi$$

$$dy=\sin\varphi dr+r\cos\varphi d\varphi$$

Substituting into A, we get

$$A=-r\sin\varphi(\cos\varphi dr-r\sin\varphi d\varphi)$$
$$+r\cos\varphi(\sin\varphi dr+r\cos\varphi d\varphi)$$
$$=r^2(\sin^2\varphi+\cos^2\varphi)d\varphi=r^2d\varphi$$

Hence $A=r^2d\varphi$ is the formula for A in the polar coordinates. In particular, we see that this is again a 1-form, a linear combination of the differentials of coordinates with functions as coefficients. Secondly, in a more conceptual way, we can define a 1-form in a domain U as a linear function on vectors at every point of U :

$$\omega(\upsilon)=\omega_1\upsilon^1+...+\omega_n\upsilon^n, \qquad (1)$$

If $\upsilon=\sum e_i\upsilon^i$, where $e_i=\partial x/\partial x^i$. Recall that the differentials of functions were defined as linear functions on vectors (at every point), and

$$dx^i(e_j)=dx^i\left(\frac{\partial x}{\partial x^j}\right)=\delta^i_j \qquad (2)$$

at every point x.

Theorem 1.9. For arbitrary 1-form ω and path γ, the integral $\int_\gamma\omega$ does not change if we change parametrization of γ provide the orientation remains the same.

Proof: Consider $\left\langle\omega(x(t)),\frac{dx}{dt'}\right\rangle$ and $\left\langle\omega(x(t(t'))),\frac{dx}{dt'}\right\rangle$ As

$$\left\langle\omega(x(t(t'))),\frac{dx}{dt'}\right\rangle=\left|\left\langle\omega(x(t(t'))),\frac{dx}{dt'}\right\rangle.\frac{dt}{dt'}\right.,$$

Let p be a rational prime and let $K=\square(\zeta_p)$. We write ζ for ζ_p or this section. Recall that K has degree $\varphi(p)=p-1$ over $\square$. We wish to show that $O_K=\square[\zeta]$. Note that ζ is a root of x^p-1, and thus is an algebraic integer; since O_K is a ring we have that $\square[\zeta]\subseteq O_K$. We give a proof without assuming unique factorization of ideals. We begin with some norm and trace computations. Let j be an integer. If j is not divisible by p, then ζ^j is a primitive p^{th} root of unity, and thus its conjugates are $\zeta,\zeta^2,...,\zeta^{p-1}$. Therefore

$$Tr_{K/\square}(\zeta^j)=\zeta+\zeta^2+...+\zeta^{p-1}=\Phi_p(\zeta)-1=-1$$

If p does divide j, then $\zeta^j=1$, so it has only the one conjugate 1, and $Tr_{K/\square}(\zeta^j)=p-1$ By linearity of the trace, we find that

$$Tr_{K/\square}(1-\zeta)=Tr_{K/\square}(1-\zeta^2)=...$$
$$=Tr_{K/\square}(1-\zeta^{p-1})=p$$

We also need to compute the norm of $1-\zeta$. For this, we use the factorization

$$x^{p-1}+x^{p-2}+...+1=\Phi_p(x)$$
$$=(x-\zeta)(x-\zeta^2)...(x-\zeta^{p-1});$$

Plugging in $x=1$ shows that

$$p=(1-\zeta)(1-\zeta^2)...(1-\zeta^{p-1})$$

Since the $(1-\zeta^j)$ are the conjugates of $(1-\zeta)$, this shows that $N_{K/\mathbb{Q}}(1-\zeta)=p$ The key result for determining the ring of integers O_K is the following.

LEMMA 1.9

$$(1-\zeta)O_K\cap\mathbb{Z}=p\mathbb{Z}$$

Proof. We saw above that p is a multiple of $(1-\zeta)$ in O_K, so the inclusion $(1-\zeta)O_K\cap\mathbb{Z}\supseteq p\mathbb{Z}$ is immediate. Suppose now that the inclusion is strict. Since $(1-\zeta)O_K\cap\mathbb{Z}$ is an ideal of $\mathbb{Z}$ containing $p\mathbb{Z}$ and $p\mathbb{Z}$ is a maximal ideal of $\mathbb{Z}$, we must have $(1-\zeta)O_K\cap\mathbb{Z}=\mathbb{Z}$ Thus we can write

$$1=\alpha(1-\zeta)$$

For some $\alpha\in O_K$. That is, $1-\zeta$ is a unit in O_K.

COROLLARY 1.1 For any $\alpha\in O_K$, $Tr_{K/\mathbb{Q}}((1-\zeta)\alpha)\in p.\mathbb{Z}$

PROOF. We have

$$\begin{aligned}Tr_{K/\mathbb{Q}}((1-\zeta)\alpha)&=\sigma_1((1-\zeta)\alpha)+...+\sigma_{p-1}((1-\zeta)\alpha)\\&=\sigma_1(1-\zeta)\sigma_1(\alpha)+...+\sigma_{p-1}(1-\zeta)\sigma_{p-1}(\alpha)\\&=(1-\zeta)\sigma_1(\alpha)+...+(1-\zeta^{p-1})\sigma_{p-1}(\alpha)\end{aligned}$$

Where the σ_i are the complex embeddings of K (which we are really viewing as automorphisms of K) with the usual ordering. Furthermore, $1-\zeta^j$ is a multiple of $1-\zeta$ in O_K for every $j\neq 0$. Thus $Tr_{K/\mathbb{Q}}(\alpha(1-\zeta))\in(1-\zeta)O_K$ Since the trace is also a rational integer.

PROPOSITION 1.4 Let p be a prime number and let $K=|\mathbb{Q}(\zeta_p)$ be the p^{th} cyclotomic field. Then $O_K=\mathbb{Z}[\zeta_p]\cong\mathbb{Z}[x]/(\Phi_p(x));$ Thus $1,\zeta_p,...,\zeta_p^{p-2}$ is an integral basis for O_K.

PROOF. Let $\alpha\in O_K$ and write $\alpha=a_0+a_1\zeta+...+a_{p-2}\zeta^{p-2}$ With $a_i\in\mathbb{Z}$. Then

$$\alpha(1-\zeta)=a_0(1-\zeta)+a_1(\zeta-\zeta^2)+...+a_{p-2}(\zeta^{p-2}-\zeta^{p-1})$$

By the linearity of the trace and our above calculations we find that $Tr_{K/\mathbb{Q}}(\alpha(1-\zeta))=pa_0$ We also have $Tr_{K/\mathbb{Q}}(\alpha(1-\zeta))\in p\mathbb{Z}$, so $a_0\in\mathbb{Z}$ Next consider the algebraic integer $(\alpha-a_0)\zeta^{-1}=a_1+a_2\zeta+...+a_{p-2}\zeta^{p-3}$; This is an algebraic integer since $\zeta^{-1}=\zeta^{p-1}$ is. The same argument as above shows that $a_1\in\mathbb{Z}$, and continuing in this way we find that all of the a_i are in $\mathbb{Z}$. This completes the proof.

Example 1.4 Let $K=\mathbb{Q}$, then the local ring $\mathbb{Z}_{(p)}$ is simply the subring of $\mathbb{Q}$ of rational numbers with denominator relatively prime to p. Note that this ring $\mathbb{Z}_{(p)}$ is not the ring $\mathbb{Z}_p$ of p-adic integers; to get $\mathbb{Z}_p$ one must complete $\mathbb{Z}_{(p)}$. The usefulness of $O_{K,p}$ comes from the fact that it has a particularly simple ideal structure. Let a be any proper ideal of $O_{K,p}$ and consider the ideal $a\cap O_K$ of O_K. We claim that $a=(a\cap O_K)O_{K,p}$; That is, that a is generated by the elements of a in $a\cap O_K$. It is clear from the definition of an ideal that $a\supseteq(a\cap O_K)O_{K,p}$. To prove the other inclusion, let α be any element of a. Then we can write $\alpha=\beta/\gamma$ where $\beta\in O_K$ and $\gamma\notin p$. In particular, $\beta\in a$ (since $\beta/\gamma\in a$ and a is an ideal), so $\beta\in O_K$ and $\gamma\notin p$. so $\beta\in a\cap O_K$. Since $1/\gamma\in O_{K,p}$, this implies that $\alpha=\beta/\gamma\in(a\cap O_K)O_{K,p}$, as claimed.We can use this fact to determine all of the ideals of $O_{K,p}$. Let a be any ideal of $O_{K,p}$ and consider the ideal factorization of $a\cap O_K$ in O_K. write it as $a\cap O_K=p^nb$ For some n and some ideal b, relatively prime to p. we claim first that $bO_{K,p}=O_{K,p}$. We now find that

$$a=(a\cap O_K)O_{K,p}=p^nbO_{K,p}=p^nO_{K,p}$$

Since $bO_{K,p}$. Thus every ideal of $O_{K,p}$ has the form $p^nO_{K,p}$ for some n; it follows immediately that $O_{K,p}$ is noetherian. It is also now clear that $p^nO_{K,p}$ is the unique non-zero prime ideal in $O_{K,p}$. Furthermore, the inclusion $O_K \mapsto O_{K,p}/pO_{K,p}$ Since $pO_{K,p}\cap O_K = p,$ this map is also surjection, since the residue class of $\alpha/\beta \in O_{K,p}$ (with $\alpha \in O_K$ and $\beta \notin p$) is the image of $\alpha\beta^{-1}$ in $O_{K/p}$, which makes sense since β is invertible in $O_{K/p}$. Thus the map is an isomorphism. In particular, it is now abundantly clear that every non-zero prime ideal of $O_{K,p}$ is maximal. To show that $O_{K,p}$ is a Dedekind domain, it remains to show that it is integrally closed in K. So let $\gamma \in K$ be a root of a polynomial with coefficients in $O_{K,p}$; write this polynomial as $x^m+\frac{\alpha_{m-1}}{\beta_{m-1}}x^{m-1}+\ldots+\frac{\alpha_0}{\beta_0}$ With $\alpha_i \in O_K$ and $\beta_i \in O_{K-p}$. Set $\beta=\beta_0\beta_1\ldots\beta_{m-1}$. Multiplying by β^m we find that $\beta\gamma$ is the root of a monic polynomial with coefficients in O_K. Thus $\beta\gamma \in O_K$; since $\beta \notin p,$ we have $\beta\gamma/\beta=\gamma\in O_{K,p}$. Thus $O_{K,p}$ is integrally close in K.

COROLLARY 1.2. Let K be a number field of degree n and let α be in O_K then $N'_{K/\square}(\alpha O_K)=\left|N_{K/\square}(\alpha)\right|$

PROOF. We assume a bit more Galois theory than usual for this proof. Assume first that $K/\square$ is Galois. Let σ be an element of $Gal(K/\square)$. It is clear that $\sigma(O_K)/\sigma(\alpha)\cong O_{K/\alpha}$; since $\sigma(O_K)=O_K$, this shows that $N'_{K/\square}(\sigma(\alpha)O_K)=N'_{K/\square}(\alpha O_K)$. Taking the product over all $\sigma\in Gal(K/\square)$, we have $N'_{K/\square}(N_{K/\square}(\alpha)O_K)=N'_{K/\square}(\alpha O_K)^n$ Since $N_{K/\square}(\alpha)$ is a rational integer and O_K is a free$\square$ -module of rank n, $O_K/N_{K/\square}(\alpha)O_K$ Will have order $N_{K/\square}(\alpha)^n$; therefore

$$N'_{K/\square}(N_{K/\square}(\alpha)O_K)=N_{K/\square}(\alpha O_K)^n$$

This completes the proof. In the general case, let L be the Galois closure of K and set $[L:K]=m.$

F. Enterprise Role Based Access Control Model

In [5] and [4] we introduced the Enterprise-Role Based Access Control Model (ERBAC) which has been implemented in the commercial security provisioning and identity management tool SAM Jupiter [10]. Enterprise Roles allow the administration of users and their access rights across all systems in the IT environment of an organisation. Enterprise Roles span over more than one target system and consist of permissions in multiple systems. These permissions are specific to the target system and can be of various natures. The example in figure 1 shows a role containing a group in UNIX, a role in Oracle and a group in RACF with authorizations for updating a dataset and reading a database table. Figure 2 shows the resulting Enterprise Role-Based Access Control model (ERBAC)2. Enterprise Roles include all permissions needed to perform a specific role. Users are then assigned to these roles. The permissions a user receives through the assignment of a role are propagated to the administered target systems (TS). The Enterprise User definition leads to the creation of user accounts (user IDs) in the TS. A permission can be any operation for an object in one of the underlying target systems. The assignment of a permission to an Enterprise Role does not necessarily cause any update in the target system. The permissions 2For a more comprehensive description of ERBAC and its comparison to the proposed NIST RBAC standard see [4]. defined for the role are propagated, and the user's accounts receive the associated permissions in the respective TS only when a role is assigned to the user. The process is the same, of course, when permissions are added to or removed from roles. In addition to the core RBAC features, a general role hierarchy is supported. Enterprise Roles can be assigned to other roles in a directed acyclic graph (DAG). Child roles inherit all permissions from their parent roles (including all permissions that these roles inherit). A user assigned to a child role thus receives all permissions assigned to this role, plus all permissions which the role inherits from its ancestors. Separation of Duty is implemented in ERBAC by rules defining constraints between roles. These rules are evaluated when assigning users to roles and connecting roles to other roles, thus preventing a user from receiving illegal combinations of roles, even in the presence of a role hierarchy.

ERBAC as described in the previous section is a proven basis for the administration of users and their access rights in medium and large enterprises. The IT infrastructures of such enterprises can consist of some ten thousand to hundred thousand users. To cope with these amounts, companies need a considerable number of administrators3. A wide range of business and system knowledge is needed to perform these administrative tasks. These are, therefore, delegated to different groups of administrators. To allow for this delegation of administrative authority, the ERBAC system itself must implement an administrative security concept. Naturally, this administrative security system is implemented as a target system itself. It uses the same entities as already defined in ERBAC. Administrators are defined as accounts in this target system and receive access rights via roles containing administrative permissions. 3Though it is possible to automate a high percentage of the administration tasks (see for example [4]), a considerable amount of manual work still remains. 4The resulting Administrative ERBAC model (A-ERBAC) is shown in figure 3. In contrast to “normal” target systems, the administrative security system is part of the ERBAC system itself, and assignments need not be propagated to some external target system. If required, Separation of Duty may be enforced by rules as described in section 2.1. The permissions for the administrative security system consist of operations allowed for the different objects in the ERBAC system. The ERBAC objects are listed in table 1. Users and roles are enterprise-wide entities, whereas accounts and permissions are specific to the target systems. The latter are distinguished for each TS because administrators are often responsible for one or more specific target systems. Relations are considered as separate objects to allow for a more fine-grained authorisation: An administrator who is responsible for building roles may not be allowed to assign users to them. Administrator accounts and administrative permissions are normal ERBAC objects. Therefore, they can be administered like all other objects by A-ERBAC. Table 2 lists the operations which can be specified in administrative permissions. All operations are valid on the object level. In addition, the View and Change operations can be restricted on the attribute level to prevent administrators from viewing or updating sensitive attributes. This is especially valuable for user and account objects. For example, we can allow administration of a RACF account but forbid change of the SPECIAL attribute providing super administrator rights in RACF. Furthermore, user attributes are often used to automate assignments of roles to users, so controlling access to them is very important (see e.g. [4] and [1]). We think that it is important to distinguish between these different operations because in real-life scenarios administrators are normally only allowed to perform specific operations. Some examples include:

• Users are inserted and deleted via an automatic connection to the human resources system. Therefore, human administrators may only be allowed to view and change users and assign roles, but not to insert or delete them.

• A typical local administrator is only allowed to assign roles to users in his department. He may view roles to see which permissions they include, but is not allowed to insert, change or delete them.

• We should not only control update access rights, but also restrict View access rights. This is important for two reasons:

– Security: principle of least privilege,

– Usability: administration is facilitated if administrators are only able to see the objects they deal with, thus reducing the amount of data they must work with.

G. Administrative Role Based Access Contol Model

The administrative role-based access control model (ARBAC97) [12] expresses the idea of using RBAC to manage RBAC through decentralisation of administrative authority, including distinction between regular and administrative roles and permissions. We do not enforce this distinction on a technical level in A-ERBAC, but agree that it is normally made on an organisational level. ARBAC97 consists of three sub-models. These describe decentralised administration through user-role assignment (URA97), permission-role assignment (PRA97) and role-role assignment (RRA97). Two central concepts of ARBAC97 are the administrative range and prerequisite conditions which regulate and impose restrictions on the administration of system objects. The administrative range reflects the set of roles over which an administrator has authority. Depending on the context, he can assign and remove users to or from a role, alter role hierarchies, and assign or revoke permissions. The authority to control user-role assignments is expressed in a relation can assign ⊆ AR × CR × 2R. For example, the expression can assign (arx, rry,{rra, rrb, rrc}), would state that a member of the administrative role arx can assign a user who currently is a member of regular role rry to the regular roles rra, rrb or rrc. With respect to such user-role assignments, a prerequisite condition could state that any user to be assigned to a role r1 must already be assigned to another role r2.

It has been demonstrated that there may be scenarios in which the decentralised administration of a system may be awkward when following the ARBAC97 approach. One example for this is the case of an external consultant assigned to the role “Employee Project X” within a project. Membership of this role might be a precondition for further assignments within the project by the local administrator. The consultant thus automatically qualifies for these possible assignments, and there is no way in URA97 to prohibit further assignments for the consultant. The ARBAC99 model [13] extends the ARBAC97 model to address such issues, introducing a notion of mobile and immobile memberships of users and permissions in roles. Immobile assignment of a user to a role allows him to make use of the rights associated with that role, however his role membership does not qualify him for any further assignments. Mobile membership on the other hand covers both aspects, access to the permissions of the role as well as the possibility of further role assignments. The problem of the external consultant could thus be easily solved by providing him with an immobile membership to the project role. Mobile and immobile assignments of permissions to roles work analogously.

The ARBAC02 model was introduced in [9], addressing a set of problems that may occur in the administration of user-role and role-permission relationships with respect to the ARBAC97 model. The first underlying reason for these problems is that in ARBAC97 the user and permission pools are dependent on the structure of the role hierarchy. Thus, the concept of an organisational unit, independent of role hierarchies, is introduced as the basis for defining user and permission pools. Assigning a user or permission to a pool is independent from assigning it to a role. The second identified reason is the top-down approach used in ARBAC97 for permission-role administration. Consequently, a bottom-up approach is suggested in ARBAC02. This means that common permissions are assigned to roles lower in a role hierarchy, while higher roles inherit these and may also be provided with other more specific permissions.

H. Policy Based Systems Management

We introduced the concept of “scope”, which appears to subsume the ARBAC02 notion of “pools”, in the context of delegation of authority [7]. In this work, scope was defined in terms of domains, which are named sets of principals and resources. The concept of domain provides a flexible and powerful mechanism for capturing many aspects of organizational structure, e.g. cost centers, or departments based on geographical or functional criteria. The security administrator’s scope of authority is constrained in two ways:

1. His “Subject Authority” limits the users to whom he can give access rights;
2. His “Target Authority” limits the resources to which he can give access rights.

The motivation for this work was the need to permit security administrators to create access rights, while not possessing such rights themselves. This is achieved by ensuring that the security administrator is not a member of the “Subject Authority” scope.

I. Generative Role Moning

The generative role mining problem can be addressed us- ing a greedy algorithm or a probabilistic approximation. In this section, we present a simple greedy algorithm for the _- distance variant which is fast but produces a large number of (unstable). Later sections describe our machine learning algorithms that are more e_cient and produce much better role assignments. Our greedy algorithm is described in Figure 1. Similar to previous greedy solutions to role mining problems [10], we begin with a set of candidate roles and select a subset optimizing the _-distance function. We make one simpli- fying assumption that users cannot be assigned roles that would authorize them for permissions they would not oth- erwise have, i.e., we do not allow over assigning permissions to users. This makes our greedy algorithm strictly conser- vative. Simply, the algorithm begins with an empty set of roles, and adds roles one at a time from the set of candidate roles, Cand, such that the next role optimizes the _-distance of the generative role mining problem. For our experiments, we use the FastMiner algorithm [20] to generate our set of candidate roles. The Score calculates the _-distance, and for simplicity, we assume it will de_ne UA such that there are no over-assignments, corresponding to _ = 1. Input: USAGE, the mapping from users and permissions to usage frequency counts; Cand, a set of candidate roles; k, the maximum number of roles to select; and _, the lambda distance for the generative role mining Output: R, an optimized subset of candidate roles Cand Algorithm:

J. Applications

Usage-based RBAC models de_ne a conservative security policy since users are assigned only those permissions which they actually use and this reduces operational risk. Gen- erative models also model exactly how users actually use the permissions. For instance generative models will distin- guish the role of a backup to an administrator who has the same entitlements but only uses them occasionally. Besides these directly apparent bene_ts, generative role models have many interesting applications which we are investigating in ongoing work.

_ Policy Reconciliation: Generative models can be used to reconcile with traditional RBAC models built from entitlements. This yields useful insights such as the evolution of role de_nitions when users begin to use some permissions more than others.

_ Identifying Policy Errors: Generative models can be used to identify a number of errors in policies such as overprovisioned users as well as users who have dif- ferent attributes than other users using the same per- misions.

_ Anomaly Detection: By comparing generative mod- els across di_erent periods of time, one can deduce changes in user behavior in terms of permission usage.

This could ag anomalous behavior such as user who starts using an entirely new set of permissions.

IV. MACHINE LEARNING MODELS

One of the key contributions of this paper is to adapt and formulate the problem of _nding generative models and vari- ants, to a family of techniques in machine learning. These generative machine learning models have been developed for unsupervised topic discovery in a large collection of docu- ments, and attempt to explain how the observations (doc- uments) were generated given certain hidden parameters. They learn the joint probability distribution between obser- vations and latent parameters and then use Bayesian mod- els to infer the parameters given these observations. This is conceptually close to the problem of explaining a set of observed logs by associating them with latent roles. In this paper we focus on the application of two widely used generative models|Latent Dirichlet Allocation (LDA) and author-topic models (ATM) [2, 15]|to generative role mining. We translate the problem of _nding latent roles to the problem of latent semantic analysis i.e. _nding the latent topics in a collection of documents. These generative models are well suited for role mining applications since they require no manual labels and allow users to have multiple roles. We provide a few details of LDA and ATM models, and how we can apply them to role mining. 4.1 Latent Dirichlet Allocation (LDA) LDA is a probabilistic generative model for collections of discrete data such as documents [2]. Each document in a corpus is modeled as a _nite mixture over underlying set of topics, and each topic is, in turn, modeled as a proba- bilistic distribution over words. LDA assumes the following generative process for creating a document d in a corpus D:

1. For each document d, a distribution over topics is sam- pled from a Dirichlet distribution, _ _ Dir(_).
2. For each word w in the document, a single topic, z, is selected according to the distribution, Multinomial(_).
3. Finally, a word is chosen from a multinomial distribu-tion over words speci_c to the topic, p(wjz; _). _ is a matrix of word probabilities over topics which is to be estimated from the training data.

Note that LDA allows an arbitrary number of topics as- signed to a document. For role mining, we can model each user's observed actions (document) as a _nite mixture over an underlying set of roles (topics) which we can estimate using LDA.

A. Author Topic Model

Author-topic model (ATM) extends LDA by adding au- thors of the documents in the modeling process and aims to simultaneously model the content of documents and the interests of authors [15,16]. ATM assumes the following pro- cess to generate a document d: For each word in d, an author is randomly chosen. Then a topic is chosen from a multi- nomial distribution over topics speci_c to the author, and the word is generated from the selected topic. Therefore, a multi-author document inherits the mixture of probabil- ity distributions associated with each author, allowing the mixture weights for di_erent topics to be determined by the document authors. An author is represented by a multinomial distribution over topics, and each topic is represented as a probability distribution over words. Assume there are T topics and W words created by A authors in a text collection. The multinomial distribution over topics for each author is pa-rameterized by _ of size T _ A, where _ta represents the probability of assigning topic t to a word generated by au- thor a. The multinomial distributions of topics over words are parameterized by _ of size W _T, where _wt represents the probability of generating word w from topic t. Author-topic models can be adapted to role mining with explicit attribution. User attributes will be the authors and as before the documents are the observed usage and topics the roles. By learning the parameters of the model, we can extract the set of topics (roles) in a corpus (USAGE), and identify which topics (roles) are generated by which authors (user attributes).

B. Performance

Our algorithms perform well on almost all data sets con- taining access logs or entitlements. For example, on a single application with 36M actions by 2050 users, our algorithm is able to produce a good stable decomposition in less than one hour. For the same application, the greedy algorithm, while faster, produces over 270K candidate roles, resulting in a slower pruning process. To compare with prior algorithms, we evaluate the performance on entitlement data against MAC, the only other state of the art probabilistic role min- ing algorithm. The biggest advantage of our algorithm is the dramatic performance improvements of several orders of magnitude. Table 2 compares running time of our algorithm with MAC over a range of datasets. We restricted the total number of roles to 25 and 15 since increasing this value will cause MAC to run unreasonably long. The key reason for the performance improvements is that MAC enumerates all possible assignments of k roles, i.e., O(2k), and then opti- mizes the assignment of permissions to the roles. Due to slow performance, MAC is often restricted to assign at most t < k roles to each user. Figure 2 illustrates the impact on restricting the maximum number of roles on the running time of MAC compared to our LDA-based approach.

C. Coverage

A set of candidate roles can be measured by how well the roles enable the users to perform their tasks (i.e., coverage). Traditionally, this is measured by the Hamming distance between the input policy and the resulting role decomposi- tion. When user access logs are taken into account, it can be better measured by the _-distance, the percentage of the access logs that can be attributed to roles. Actions which users perform infrequently or performed by a small number of users may be exceptions, permissions directly assigned to the users, or delegation.

We compare the coverage of our generative models (dLDA) with our greedy algorithm and MAC across a num-ber of application logs and six-month time windows drawn from the source code repository logs. Table 3 shows the num- ber of applications in each time window, and the number of applications that achieve high levels of coverage from 80% to 99% using dLDA. As we can see, dLDA have very good cov- erage across most applications and time periods achieving the _ 90% coverage level for almost all applications. Table 4 shows more detailed performance for a speci_c time period across all applications. The names of each ap-plication have been anonymized. Note that our algorithm does produce a small number of over-assignments. The table presents the over-assignments and coverage instead of pre- senting the uni_ed _-distance. As can be seen, the speci_c results show even better coverage, with some applications more than 99%. A small number of applications, however, do not have good coverage due to insu_cient data. See Ap-pendix A. In general, we can increase the performance by increasing the size from which we draw usage data. We have also compared our results with MAC and our greedy algorithm based on FastMiner to generate candidate roles. First, we produce a binary UP relation from the USAGE relation, such that we assign a user a permission if they used it at least once. The MAC algorithm is applied to the binary relation, and, using the USAGE relation, we calculate the coverage as the _-distance for _ = 0. For our greedy algorithm, we apply the FastMiner algorithm on the binary UP relation to produce a set of candidate roles. We then apply our greedy algorithm such that there are no over assignments. Note that the larger the candidate set of roles, the tighter we expect the coverage to be. The results are shown in Table 5. We can clearly see that all three algorithms adequately cover the permission usage logs, but each has a trade o_. Some assignments may have been used a small number of times by a small clus-ter of users, resulting in MAC de_ning a role for these as-signments. Our generative approach did not recognize the infrequent usage as a role, resulting in slightly decreased coverage and more under assignments, but also fewer over assignments. Many infrequent operations are for sensitive operations, and should not be over assigned. We do _nd that the greedy algorithm outperforms both our generative role mining algorithm and MAC on coverage in the major- ity of the example datasets. This is not surprising, and we would expect any role mining algorithm that speci_cally op-timizes a _tness function, such as WSC or _-distance, will. However, it is not clear that the resulting roles are mean- ingful, and often represent infrequently used permissions. In the next section, we will illustrate that the roles from both MAC and our greedy algorithm produce roles that are unstable|they are over _t to the data observed in each six month time period and must undergo signi_cant alterations in each time period to perform consistently.

D. Mining Entitlement Data

We have also compared our generative role mining algo- rithm with MAC on entitlement data. To mimic permis- sion usage logs, we assign a default value, w to each user- permission pair, and then apply our generative role mining algorithm as usual. In Section 6.3 we discuss the impact of w. Here, we compare each algorithm's ability to reconstruct the original input user-permission relation and measure the normalized Hamming distance for varying values of k. For performance reasons, we restrict MAC to at most two roles per user. We later relaxed this restriction to three without a signi_cant improvement. Due to the running time of MAC, we were not able to relax the restriction further. The _nal results for varying values of k are shown in Figure 3. This _gure clearly illustrate the performance of our generative approach is comparable to MAC for entitlement data, and even outperforms MAC on the Customer dataset.

E. Stability

A crucial assumption in RBAC is that the permissions as- signed to a role and the basic role structure should be largely static over time, while the users assigned to the roles (and implicitly how much the users use each role) may change over time. This reduces the administrative complexity, and underlies the intuition behind weighted structural complex- ity measures [12] used throughout the role mining litera- ture [4, 10, 19]. We evaluate how well our log-based role mining algorithm produces static roles by measuring the sta- bility or consistency of the permissions assigned to the role when mined from di_erent time periods. For each six-month period, we run our dLDA algorithm to produce a set of roles, and measure the stability as a maximal matching between roles in one set with roles in a second set as described in- dependently by Molloy et al. [13] and Frank et al. [6]. By _nding the closest one-to-one matching between the roles, we calculate how dissimilar the matched roles are using a distance function, such as the Jaccard distance. The more dissimilar, the less stable the roles are. We also compare the probabilistic algorithms against a more traditional role mining algorithm, FastMiner, when restricting FastMiner to the same _xed number of roles and applying our greedy al- gorithm to maximize the _-distance. Given n six-month time periods, we mine the roles in each time period, and calculate the dissimilarity for all n2□1 role- set pairs. All scores are normalized for each role. The results are plotted as histograms indicating the overall performance of the algorithm for the given dataset, and a normal distri-bution is _tted to the data as shown in Figure 4. It can clearly be seen that the roles produced by our generative algorithm are more stable across all time periods, i.e., re- quiring few changes over a _ve year period. This is a key property the roles should have to ensure their adoption and continued use.

V. MINING ROLE BASED ACCESS CONTROL POLICIES

For role mining with explicit attribution, we use the Author-Topic model which extends the models of LDA. We assume that besides the user-permission data, we are also given a list of attribute values for each user. The goal is to _nd a role decomposition which is correlated with the attributes of the user. The translation of this problem to ATM is again straightforward: As before, the words are the individual permissions, the documents are the users (permis- sions assigned to the users). In addition, the authors are the attributes of the individual users. Abstractly, applying this model to access control logs pro- duces the following analogy describing their creation. First, an attribute, or set of attributes, are selected that de_ne some job function of the user. From these attributes, we se- lect a role through which the user will act, and will provision them the necessary permissions to function in the job. Fi- nally, given the role (which is selected solely from the user's attributes), we select an action the user will perform. Suc- cinctly, it is the attributes which entitle the user's to roles, and thus permissions. ATM can thus be used to obtain a distribution from au- thors to topics, i.e., from the user attributes to the assigned roles, in addition to the distribution _, from topics to words. This yields a role decomposition which has explicit attribu- tion. As before, we will need to discretize these distributions to explicit role assignments for each user. For a given user, we average the probability distribution corresponding to the each of the attributes of this user. This yields a probability distribution over assigned roles which can be discretized as described in Section 5.2.

A. Preprocessing

Very crucial to the performance of ATM is the choice of relevant user attributes to use in the model, as well as cleans- ing the values of these attributes. First, attribute values need to be inspected to rationalize di_erent values which are semantically the same (e.g., country = USA and coun- try = US). Further, using all user attributes results in poor performance and greatly increase the time to _t the model to the data. A simple measure to identify relevant attributes is to discard any attribute value which is not assigned to more than a threshold number of users [7, 13].

B. Arbitrary Attributes

In contrast to prior approaches [7], we want to allow any arbitrary number of attributes to imply a role. In particular, using the kernel-trick from machine learning, we precom- pute attributes which are boolean functions of the original attributes and use these as the authors in ATM. In our im- plementation, we use a few optimizations: Typically in the provisioning of entitlements, one never uses the negation of an attribute to provision roles. In the author-topic model, attributes will be selected uniformly, and any attributes may imply the assignment of a given role. As a result, role as- signment through disjunction is provided \for free", Thus the only boolean functions we need to consider are only con- junctions as disjunctions arise naturally from the de_nition of the ATM. As a further optimization, we only consider conjunctions with at most three conjuncts since we believe that for larger conjuncts it is more natural to de_ne a new attribute reecting these larger conjunctions. This is not a limitation but simply a design choice. Ours is the _rst to allow for both disjunctions and conjunctions of attributes to imply permissions and roles.

C. Examples

Let us consider an example to show how the TrustBAC framework works. For this purpose we assume that a digital library system DL uses TrustBAC to control access privileges of its users for the resource present in that DL. The DL has an access control policy ADL and a trust evaluation policy TDL. Let basic user and privilege user be two roles in the ROLES set of the digital library. We assume ADL specifies the following: Assigned Roles([0.05, 0.4]) = basicuser and Assigned Roles([0.35, 0.6]) = privilegeuser. Let a user u log in to the system and manifest a set of credentials c (for simplicity we assume that user properties are expressed in terms of credentials). The system initiates a session sc and the trust relationship (DL −→c u)Nt is considered. The credentials are verified and evaluated and the corresponding value is stored in DLKcu . The session history uhc is consulted and the trust is evaluated as v(DL −→c u)Nt = 0.45.

Therefore, according to Assigned Roles the user at this stage is allowed to act as a privilege user as well as a basic user. Let the user select the role of privilege user. Let the privilege users of DL be allowed to write comment about the articles present in the database as well as can upload digital copies of articles that are not present in the database. Let the system consider abusive/irrelevant comments as negative events and upload of a corrupted or inauthentic file as negative event. Let u during the session sc write several bad comments and upload a few inauthentic files. Each of these activities get reported in the session history uhc. To handle recommendation we assume that DL is a part of a digital library consortium where the member DLs are linked to one another. During the session the DL system sends messages requesting for recommendation from other members of the consortium about u. Let TB evaluate trust periodically within a session.Let at some evaluation point v(DL −→c u)Nt = 0.345. This shows that u is no longer 'trustworthy' to the system as a privilege user. The system automatically withold the role of privilege user for u. During the remaining time in this session u can no longer act as a privilege user. So if there is a section of articles in the DL which is only available to privilege users then u can not access those articles anymore. However, u can continue to act as a basic user. Otherwise she may logout. The next time u logs in with properties c, u can only perform the role of a basic user. Good actions and good recommendations can increase the trust level for u and when it reaches 0.35, u is again able to act as privilege user. Alternatively, u can produce some extra credential (something like a special permission from the digital library authority) in the new session to raise her trust level. However, the set of extra credential alone may not be sufficient to raise the trust level. u may still need to behave well. How the trust level decreases for bad behavior or increases with extra set of credentials depends on TDL. u can deliberately perform malicious actions as a privilege user to get personal benefit without caring about her trust level. For example she may want to decrease the rating of a article written by someone she hates by putting bad comments about it. When she is restricted to perform as a basic user only then she starts behaving well to increase her trust level to act again as a privilege user and repeats the cycle. To prevent this type building trust and then milking the system can be prevented by "slow-to-increase' and 'fast-to-decrease' policy. The TDL can be so configured that every bad action is heavily penalized to lower the trust level rapidly. Every good action adds only a little amount to the trust level. So it will either need extra credentials and set of good recommendation or consistent good behavior over a series of sessions.

D. Authorization Rules

Authorization rules are a set of requirements that should be satisfied before allowing subjects' access to objects or use of objects. There exist two kinds of authorization rules. They are Rights-related Authorization Rules (RAR) and Obligation-related Authorization Rules (OAR). The RAR is used to check if a subject has valid privilege to exercise certain rights on a digital object. Examples include identities or roles verification, capabilities or properties checking, proof of payments, etc. The OAR is used to check if a subject has agreed on the fulfillment of an obligation which has to be done after obtaining or exercising rights on a digital object. Examples include metered payment agreement, usage log report agreement, etc. The authorization rules are different from conditions. The authorization rules are a set of decision factors used to check whether a subject is qualified for the use of certain rights on an object, whereas the condition is used to check whether existing limitations and status of usage rights on an object are valid and whether those limitations have to be updated.

E. Conditions

Conditions are a set of decision factors that the system should verify at authorization process along with authorization rules before allowing usage of rights on a digital object. There are two types of conditions: Dynamic conditions and Static conditions. Dynamic conditions include information that may have to be checked for updates at each time of usage. Static conditions include information that does not have to be checked for updates. Dynamic conditions are stateful and the static conditions are stateless. Some examples of dynamic conditions are the number of usage times (e.g., can read 5 times, can print 2 times), and usage log (e.g., already read portion cannot be accessed again). Some examples of static conditions are accessible time period (e.g., business hours), accessible location (e.g., workplace), and allowed printer name.

F. Obligations

Obligations are mandatory requirements that a subject has to perform after obtaining or exercising rights on an object. In real world implementation, however, this may have to be done by agreeing on the fulfillment of obligations before obtaining the rights and at the time obligation-related authorization rules are checked. For example, a consumer subject may have to accept metered payment agreements before obtaining the rights for the usage of certain digital information or should agree on providing usage log information to a provider subject before reading an ebook or listening a music file. Traditional access control has hardly recognized the obligation concept. Recent DRM solutions are likely to include obligation functions though many of them implement the obligation functions only partially and implicitly.

VI. SEMANTICS

In the following, we provide the formal semantics for an ACMI. It has been shown that any C-Datalog program can be transformed into an equivalent Datalog program with negation [Greco et al. 1992]. Given an ACMI I, we denote with D(I) the corresponding Datalog-like program, that we call Access Control Model Program (ACMP). The interested reader can find some information concerning this transformation in Appendix A. Moreover, given an ACMS S D< B, Scheme, ISA, A, Z >, we denote with L(S) the logical language over which program D(I) is constructed, where I is an instance of S, and we call it Access Control Model Language (ACML). It is simple to prove that constants in L(S) coincide with Z and predicates coincide with B. In the following, when S is not relevant or it is clear from the context, we denote L(S) with L. The semantics we propose has to cope with conflicting authorizations. More precisely, a conflict arises when a positive and a negative authorization hold for the same subject, object, and privilege. Conflicts have to be solved to determine whether an access should be authorized or not. The proposed semantics supports a parametric conflict resolution policy that establishes which authorization prevails possibly exploiting information about the authorization sources. The exact conflict resolution policy depends on the access control model being modeled. In the following, we first deal with the problem of conflicts and we then present a formal semantics for ACMIs.

A. Conflict Management

Let D(I) be an ACMP. With D(I)ground we denote the set of all ground rules of D(I) obtained by replacing each variable appearing in a rule of D(I) with a constant of the “right” type. Conflicts and conflicting rules are defined as follows.

Definition 3. Let D(I) be anACMPover L. Two atoms A1, A2 are conflicting in L if A1

It is well known that there is no unique solution to the problem of conflict management and that several conflict resolution policies can be defined depending on the specific domain [Ferrari and Thuraisingham 2000]. Examples of conflict resolution policies are denials take precedence, most specific authorization takes precedence, and permissions take precedence. In order to provide a flexible conflict resolution mechanism, a parametric conflict resolution policy is introduced that, for each conflict, specifies how the conflict has to be resolved, possibly also taking into account the authorization sources.

Definition 4 (Conflict Resolution Policy). Let L be an ACML. A conflict resolution policy for L (denoted by FL) is a total function from conflicts(L) to fC, ¡g. Given a conflict c, the intended meaning of FL(c) is to choose whether the positive or the negative authorization in c should prevail.

B. Model – Theoretic Semantics for ACMIs

ACMPs are logic programs with (arbitrary) negation. Since we make no restriction on the type of negation, we know, from logic programming, that a single meaning cannot be always assigned to these programs. This means that, in general, an ACMP is associated with different sets of entailed authorizations. The most general semantics for logic programs with negation is the stable model semantics [Ullman 1989]. This semantics assigns to a logic program a number (possibly zero) of alternative models,8 each representing a set of consistent authorizations that can be possibly assigned to subjects. In the following, we propose a stable model semantics for ACMIs. Most of the notions we introduce in the following are classical logic programming concepts. However, we need to extend the classical stable model semantics to deal with conflicts. Before presenting the proposed semantics, some preliminary definitions have to be given. Let D(I) be an ACMP over L. The base BD(I) of L is the set of all ground atoms that can be constructed from predicate symbols in B and constants in Z. A set of ground atoms is consistent if it does not contain any conflicting atom. An interpretation I for D(I) is any consistent subset of BD(I). Let I be an interpretation for D(I), L a ground literal, and r D H Ã A1, : : : , An, not B1, : : : , not Bm a ground rule. Then, L (not L) is true with respect to I if L 2 I (L 62 I); the body of r is true in I if all its ground literals are true in I . An authorization rule r 2 D(I)ground is true in I if either its head is true in I or its body is not true in I . In traditional logic programming, a model is simply defined as an interpretation in which all program rules are true. This notion is not sufficient in our context since we have to deal with conflicts and to ensure that the model does not contain conflicting atoms. This is possible by not considering all the rule instances that lead to some conflicts. This notion is formalized by the concept of discarded rule.

C. NIST Model

In the following, we show how the general framework for modeling role based access control (RBAC) models proposed by Sandhu et al. [2000] can be increasing complexity such that each level adds to the previous one new features. These levels are described in the following.

Flat RBAC. Flat RBAC is the base level, able to capture the basic classical features of an RBAC model: users acquire permissions from roles; a user can be assigned to many roles and a role can refer to many users (the same holds for the relation existing between permissions and roles); users can simultaneously exercise permissions deriving from different roles. Additionally, Flat RBAC supports user-role review, that is, it must be possible to determine which roles are assigned to a specific user and which are the users authorized to play a specific role.

Hierarchical RBAC. Hierarchical RBAC adds to Flat RBAC the support for role hierarchies. Two different interpretations of role hierarchies are supported: the inheritance and the activation interpretation. In the first case, the activation of a role ri implies the activations of all roles r j that are less powerful than ri and thus the inheritance of their permissions whereas, in the second case junior roles must be explicitly activated.

ConstrainedRBAC. Constrained RBAC adds to Hierarchical RBAC the support for separation of duty (SOD) constraints. Separation of duty is the ability to state which roles cannot be simultaneously assigned to the same user (static SOD) or which roles cannot be activated together by the same user (dynamic SOD).

Symmetric RBAC. Symmetric RBAC adds to Constrained RBAC support for permission-role review. This is the ability to determine which are the roles to which a particular permission is assigned as well as which are the permissions assigned to a particular role. The basic components of the NIST model can be formally defined as follows: —U, R, P, and S represent respectively the sets of users, roles, permissions, and sessions. Each permission is a pair (a, o) and represents a specific access mode a on object o.We thus denote with A and O the sets of access modes and objects, respectively. Thus, P µ A£ O. Moreover, let p 2 P be a permission, In order to show how each of the above models can be represented by our framework, we show that, for each considered NIST level, an access control model instance exists such that its stable model exactly represents the set of access authorizations entailed by the considered access control model. For simplicity, in presenting the mapping, differently from the NIST model, we assume that Flat RBAC also supports the notion of session. In Flat RBAC, we assume that during a session all the roles the user is authorized to play are activated. In the other cases, we assume that the user can activate a subset of the roles he/she is authorized to play.

Flat RBAC. In Flat RBAC users, roles, and permissions are flat domains; users and permissions are assigned to roles. Permissions are always positive. The ACMS and the ACMI for Flat RBAC can be constructed as follows.

D. Inter-Model Properties

Inter-model properties define the different dimensions that can be used to compare two access control models. The first comparison dimension concerns the modeling capabilities of the models. By modeling capabilities we mean all constructs provided by an authorization model to represent subjects, objects, and privileges. Examples of modeling capabilities include roles, groups, and negative/ positive privileges. Another way of comparing two access control models is on the basis of the authorizations they enforce, independently from the possibly generated errors. The last considered comparison dimension concerns consistency of the models. In the following, we introduce these properties, presenting some examples and formally discussing decidability results.

Structural Subsumption=Equivalence. Consider two models, one supporting authorizations on groups and the other supporting only the specification of authorizations for single users. These models could entail the same set of user authorizations. However, their expressive power is not the same since the first supports groups, whereas the other does not. These considerations suggest us to consider a dimension, that we call structural subsumption/ equivalence, that verifies whether two access control models are built from the same set of ACMS basic components. Two aspects have to be considered when dealing with structural equivalence. The first concerns the components contained in the ACMS for the considered access control models. For example, if an access control model deals with groups, and therefore requires class names group and user, whereas another access control model only deals with users, thus requiring only class name user, the two access control models are not structurally equivalent. In this case, we say that the access control models are not weakly structurally equivalent. The second aspect concerns the attributes used to characterize subjects, objects, and privileges. For example, a mandatory access control model assigning to each subject and object an access class is structurally different from an access control model which does not consider this information. In this case, we say that the access control models are not strongly structurally equivalent. Weak and strong structural subsumption/equivalence can be formally defined as follows.

Example 4. Consider two different role-based access control models, say Model1 and Model2, whose class entity names schema and corresponding IACMI are presented in Figure 9 and Figure 10, respectively. The main role a boolean attribute, temp dis, that is true when the role is temporarily not usable by the users authorized to play it. This means that when temp dis is true the privileges assigned to that role are not propagated to the users authorized to play it. Moreover, Model2 contains a user-defined predicate, PropDir, that, for each role and privilege, specifies whether the privilege is propagated according to either an upward or downward direction in the role hierarchy starting from that role. From Definition 12, it follows that Model1 and Model2 are w-structurally equivalent. Indeed, since user-defined predicates are not taken into account, the corresponding ACMSs are based on the same class names. On the other hand, they are not s-structurally equivalent, since class role has two different sets of attributes in the two models. However, Model1 is s-structurally subsumed by Model2, since all the attributes defining classes in Model1 are also contained in the schema of the corresponding classes in Model2. Weak (strong) structural subsumption/equivalence is of course always decidable, as stated by the following theorem, since it corresponds to determining inclusion between finite sets.

Access Subsumption/Equivalence. Two access control instances are equivalent if they enforce exactly the same set of accesses. We call this kind of equivalence access equivalence. On the other hand, an access control instance is access-subsumed by another access control instance if the set of accesses enforced by the first instance is also enforced by the second one. For both these dimensions we can consider more than one version depending on the "granularity" of the sets of accesses we compare: a strong version compares all the accesses, a weaker version compares only sets of positive accesses, whereas the weakest version compares only sets of positive accesses where the subject is a user. Subsumption and equivalence can be first analyzed with respect to access control model instances, resulting in the following definition.

E. Intra-Model Properties

Intra-model properties concern the analysis of the characteristics of a single access control model. In analyzing access control models, we have devised the following set of relevant properties:

—Reachability: by reachability we mean the ability to determine whether a certain authorization can be generated by a given access control model, possibly conditionally to the generation of another authorization. This property can be useful for determining dependencies existing among authorizations.

—Consistency: An access control model is consistent if it admits at least one instance that satisfies all the specified constraints, that is, it generates at least one consistent set of authorizations. The previous properties can be formally defined in our framework as follows.

F. CBAC Models

Our approach to constructing a family of access control models is based on the comprehensive coalition domain model described in Section 3. Distinctions between the domain models focus on the inclusion (or exclusion) of the Teams and Tasks entities within the domain model. The basic model includes neither. Two additional models include either Teams or Tasks, but not both. Finally, the most complex of the models includes both Teams and Tasks. These four coalition domain models underlie the CBAC family of access control models, named: CBACbasic, CBACteam, CBACtask, and CBACteam+task. In this section, we sketch some of the formalisms defining the CBAC domain models. An instance of any one of the domain models gives a collection of sets, where each set corresponds to one of the boxes in Figures 3.1, 2, or 3. The instance also gives several relations, each of which corresponds to one of the arrows in the figures. The domain model imposes several constraints on these relations, which serve to ensure that the relationships between the domain entities are coherent and meaningful. Space constraints limit our treatment of the domain models, however their complete formal definitions can be found in [3]. We begin by defining the domain model for each member of the CBAC family and identifying a few illustrative relationships among the domain entities, as well as some of the constraints on those relationships. After describing each of the four domain models, we develop the access control models. In particular, we introduce the concepts of "authorization set" and "protection state," and identify necessary constraints. In the model definitions below, we make use of some additional notational conventions. Relations among domain entities are represented below in a bold font and parameters and bound variables are shown below in a sans serif font. P(X) is used to denote the power set of a set X, that is to say, the set having as its members all subsets of X. A dot "." is a scoping notation and stands for a left bracket whose mate is as far to the right as is possible without altering the pairing of left and right brackets already present.

G. CBAC Domain Model

CBACbasic is built on the simplest domain model in the CBAC family, adding coalition entities to a simple RBAC model. It includes the domain entities Coalitions, PartnerOrganizations, Missions, PrincipalFunctions, OrganizationAssets, CoI, Organizations, Functions, OrganizationResources, Roles and Users and relations among them. For example: ² ParticipatesIn: PartnerOrganizations £ Coalitions holds for (p, c) when p is a member of c.

Example: The US Army has signed a memorandum of agreement with FEMA to participate in Project Impact. Thus, (Army, Project Impact) 2 ParticipatesIn ² AccomplishedBy: Missions ! Coalitions maps each mission to the coalition that has accepted it. Example: One of the specific Missions recently undertaken by Project Impact is the repair of the Pajaro River levee in northern California. Thus, AccomplishedBy(Pajaro River Flood Mitigation) = Project Impact ² MissionPartner: Missions £ PartnerOrganizations, holds for (m, p) when p is involved in the execution of m. The consistency of ParticipatesIn, AccomplishedBy and MissionPartner is assured by the following constraint, which states that any partner in a mission participates in the coalition which is to accomplish that mission: 8m 2 Missions 8p 2 PartnerOrganizations . MissionPartner(m, p)) ParticipatesIn(p, AccomplishedBy(m)) Example: This constraint ensures that when we say that the Army is a partner in the Pajaro River Flood Mitigation mission, then we can be sure that the Army is participating in Project Impact, which is accomplishing the Pajaro River Flood Mitigation mission. By grouping together users acting in roles, involved in coalition activities, CBACbasic provides important links between the individual users of the operations view and the organizations of the organization and coalition views. However, this relatively simple framework for modeling relationships among users and their organizations offers very little support for identifying relationships between collections of users or their activities. To better support such relationships, we developed the CBACteam, CBACtask, and CBACteam+task models, presented below.

H. CBAC Policy and Administration

The CBAC family of models offers a selection of paradigms for coalition-focused access control. To express access policies that have CBAC-based semantics, we are in the process of developing a policy expression language. To write access policies that are acceptable to coalition participants, we also need an administrative model for CBAC. An administrative model specifies how policy is developed and maintained. In particular, it answers questions regarding which entities are authorized to specify policy for which resources. See [14], for example, in which an administrative model was developed for RBAC. Within an organization that commonly practices delegation of authority along the organizational hierarchy, an administrative model might assume that a high-level administrator (i.e., an administrator representing the highest level of the organization) has broad authority to specify access policy for all resources “owned” by the organization and may delegate authority for some policy specification to lower-level administrators. Such a hierarchical model does not extend well to a coalition environment, as coalitions do not include a top-level organization in which ultimate authority is vested. In coalitions, administrative authority must be somewhat distributed among the partner organizations and that distribution may be negotiated. Partner organizations may use more hierarchical administrative models internally. An administrative model that supports both hierarchical and distributed delegation of authority will not be trivial to develop. Consider, for example, an organization A that participates in coalition C. A-users have authorizations to coalition resources and A-administrators define roles and assign users in roles to teams that operate within the coalition. In defining roles and assigning users to those roles, Aadministrators delegate authority within their own organization. In authorizing access to a resource by coalition teams (some of whose members may be employed by other organizations), A-administrators are delegating some authority over those authorizations to the organizations employing the team members. If A is ejected from the coalition, it is probably necessary to remove all authorizations held by A-users from authorization sets for any of the coalition resources. Must role-based or team-based authorizations be eliminated as well? If so, which entities are authorized to perform this policy change? The development of acceptable answers to such complex administrative questions is critical to the effectiveness of information resource sharing within coalitions. Such issues will require significant future research.

I. Implementation Complexity

The implementation of some CBAC models by access policy enforcement mechanisms may prove challenging. In particular, system context information, including user sessions and the activation states of missions, functions, tasks, etc., may be complex to implement. The family of CBAC models was developed to enable tradeoffs between expressive power and implementation complexity. We expect that, in the simplest case, a context-free variant of CBACbasic could be implemented with defined missions, functions, teams, and roles in a perpetually active state, eliminating the need to manage activation states. On the other hand, the implementation of models supporting teams or tasks poses a variety of implementation challenges. We group these challenges into three categories: those that arise from the distributed computing environment, those that arise from active management of authorizations, and those that arise from the need for organizational autonomy within coalitions. The first category of implementation challenges results from the complexity of distributed computing. For example, events that trigger changes in task and team state may originate from information systems in different partner organizations. Detecting and correlating those events to identify state changes is necessary to monitor task progress. The often tightly coupled events of workflow systems exacerbate the problems of identifying and maintaining a consistent view of task state within a distributed system. [12] discusses many of the issues involved with integrating access control into a distributed, inter-organizational workflow environment. The second category of implementation challenges originates from the need to support active management of authorizations. For example, both the CBACteam and CBACtask models call for state-based activation and deactivation of permissions in accordance with the current state of tasks and teams, respectively. A delay (or failure) in the granting of an authorization may result in the rescheduling of certain workflow tasks. Conversely, the failure of a task or its subtasks may result in certain permissions not being activated and granted. Thus, the liveness and termination properties of distributed state management systems may impact the robustness and safety of active authorization management. The third category of implementation challenges results from the need to support organizational autonomy within coalitions. A variety of challenges fall into this category, among them, the need for trust management and distributed delegation of authority. Collaborating, autonomous organizations are likely to want and need to cede some authority to one another. For instance, one organization may need to trust another concerning which employees it assigns to various roles. Systems to support delegation of authority and specification and evaluation of trust policies [2, 13], as well as systems to distribute and enforce coalition access control policies [16, 17] will be needed. Overall, a complete implementation of the CBAC access control models will involve the integration of security mechanisms with a wide variety of platforms, network infrastructures, workflow applications, and distributed computing systems across organizational boundaries. Much of the complexity involved in implementing CBAC models arises from the inherent complexity of coalition-based interactions. Implementations of other access control models face the same complex issues, however, when they are used within distributed coalition environments. Consider, for example, a coalition that uses an RBAC access control model to coordinate members' activities toward joint objectives. Roles, constraints, hierarchies, etc. must be used to encode any necessary coordination information. Thus role names such as Pajaro River hydrologist or Pajaro River engineer may be used to indicate a team like relationship. Of course, such roles must be activated at the appropriate times (e.g., simultaneously, or in a specific sequence) to support a collaborative application. Thus, role activation must be managed according to the state of the distributed system. In a coalition using RBAC, the requirement for such distributed state-based activation management is implicit, falling to the distributed system infrastructure. In general, access control for coalition environments places complex, implicit requirements on implementations. By directly codifying essential aspects of coalition-focused access control (e.g., organizational autonomy, negotiated agreements, shared responsibilities), the CBAC models capture explicit implementation requirements. This may improve the chances that implementations will meet the requirements.

VII.ABSTRACT ACCESS CONTROL MODEL

An abstract access control model is comprised of abstract tokens and abstract operators. Abstract tokens are assigned to RDF triples through authorization rules, whereas abstract operators describe (i) the computation of access labels for implied triples and (ii) the propagation of access labels along the RDFS class and property hierarchies. RDF triples are either annotated with tokens or with a complex expression that involves the tokens and operators of the abstract model. In our work annotated RDF triples are represented as quadruples. A quadruple is of the form (s; p; o; l) where s; p; o are the subject, property and object of the triple and l is an abstract access control expression. Now we are ready to de_ne the notion of access control model.

Definition 4.1. An abstract access control model M is a tuple M= hL;?;_; i where:
_ L is the set of abstract access tokens _ ? is the default access token that is assigned to triples that are not in the scope of some authorization rule _ _ is the binary inference operator The access label of a quadruple is an expression that is de_ned over the abstract tokens in L and the inference and propagation operators.

A. Conflict Resolution

We distinguish between two levels of conict resolution.

1. Simple Conict Resolution
2. Inference Conict Resolution

2.3.1 Simple conflict resolution

Intuitively we require that subsuming patterns have less restrictive security classi_cations than the more speci_c, sub- sumed patterns. The intuition behind this policy is that general patterns can de_ne access restrictions on a set of statements, while exceptions can be represented by the more speci_c patterns. Based on the \more restrictive takes prece- dence" resolution, the exception will be correctly classi_ed at the higher security level. Algorithm 1 addresses these issues. Simple conict resolution addresses the problem that there might be several RDF-patterns that can be mapped to a particular RDF/S statement. This could result in di_erent security labels for the same RDF statement.

Clearly, this is undesired. In this case, we choose the most restrictive clas- si_cation or the lowest upper bound of the security labels that can be assigned to the statement.

B. Inference Conflict Resolution

The second level conict resolution, called Inference con- ict resolution, addresses potential inconsistencies that oc- cur due to newly entailed RDF/S statements. Table 2 shows the automatically assigned security classi_cations to the en- tailed statements. However, it may occur that the generated triple may already be in the security cover with a di_er- ent security classi_cation. It may be a security violation if the existing triple has a higher security level; i.e.,a triple classi_ed at higher level can be inferred from lower secu- rity triples. Also security patterns from the policy may also be mapped to the newly generated statement resulting in a di_erent security label. This scenario may cause a security violation if the policy requires that the triple be classi_ed at a higher level than the level generated by the inference process. These conicts are resolved as de_ned below. Al- gorithm 2 includes this inference conict resolution as part of the security label generation for the entailed data triple. Let t be the entailed triple after applying an inference rule and sl its generated security classi_cation, i.e., there is an entailed security object (t; sl).

Case 1:If t already exists in S such that (t; sl0) 2 S then a If sl0 > sl, then there is a security violation and gen- erate the inference warning. b If sl0 _ sl, then keep the existing pair (t; sl0) in the security cover(i.e., do nothing).

Case 2: If there is no (t,sl') in S, but there exists a map- ping from security policy to the generated triple (De_nition 2.6) and the mapped security object is (t; sl0) then a If sl0 > sl then there is an inference violation and a security warning is generated. b If sl0 _ sl then discard (t; sl) and add (t; sl0) to the security cover S, i.e., not over-classify the triple. Intuitively our conict resolution considers the following options:

1.a The label of the generated triple is strictly dominated by the security label of the existing triple. This rep-resents an inference security violation since a higher security triple can be inferred from lower security la-bels.

1.b The entailed triple has higher security label than the existing one. This may occur if high security data is used to infer data at a lower level. This does not represent a security problem because the lower security label of the existing triple comes from the policy mapping and hence is safe. So we keep the existing classi_cation.

2.a The automatically generated security label is strictly dominated by the security label of the mapped RDF- pattern. This is a security violation via unauthorized inference. If t would have been in the original RDF/S (extensional) data, this case falls back to Case 1.a.

2.b The security label of the newly generated triple dom- inates the security label of the mapped RDF-pattern. This does not represent any security problem since the entailed triple is classi_ed at a higher security level than the policy require it to be. Hence the entailed triple is assigned the security label of the mapped pat- terns to avoid the over classi_cation.

C. Obligation Model

Typical obligations in privacy policies specify what actions a subject must perform at certain time in order to allow certain actions to be taken at present. Before presenting our obligation model, we first investigate the usage of obligations in privacy policies using a few case studies. We design our obligation model based on the analysis of these scenarios.

One of the regulations in COPPA requires that “Before collecting, using or disclosing personal information from a child, an operator must obtain verifiable parental consent from the child’s parent. This means that an operator must make reasonable efforts (taking into consideration available technology) to ensure that before personal information is collected from a child, a parent of the child receives notice of the operator’s information practices and consents to those practices.” Thus, an obligation, “notifying a parent and obtaining verifiable parental consent”, may have to be fulfilled before access to the children information. Another interesting point here is that we may need a conditional obligation. Once we have fulfilled the obligations and obtained some results, either consent or rejection, from a specific parent, we usually should not ask the parent the same question again. Therefore, before fulfilling the obligation, we may want to check whether we have already asked the question. Another example that requires a conditional obligation is: “An operator is required to send a new notice and request for consent to parents if there are material changes in the collection, use or disclosure practices to which the parent had previously agreed.” Sending a new notice and requesting for consent can be considered as conditional obligations after collecting children information. COPPA also says that “At any time, a parent may revoke his/her consent, refuse to allow an operator to further use or collect their child’s personal information, and direct the operator to delete the information.”. Our understanding of this statement is that once operators obtain parental consent and collect children information, they should immediately assign parents permissions such as “revoke consent”, “refuse further use or collection”, or “request deletion”. One way to specify those permissions in formal policies is to consider them as obligations that should be fulfilled without delay after children information collection. Therefore, obligations may include actions like “grant permissions”, “revoke permissions” and “delete data”. The GLB act says that “Consumers are entitled to receive a privacy notice from a financial institution only if the company shares the consumers’ information with companies not affiliated with it, with some exceptions. Customers must receive a notice every year for as long as the customer relationship lasts.”. An obligation, “sending consumers a privacy notice”, should be fulfilled after the first time consumers’ information is disclosed. Another obligation, “sending customers privacy notices”, must be fulfilled periodically. Moreover, the latter, “sending customers privacy notices”, seems not to be related to any action on customers’ information. However, such an obligation is related to an attribute of information providers. The attribute specifies that these providers are not only consumers, but also customers1. Therefore, the latter obligation is actually related to an action that changes a consumer to a customer. Based on aforementioned cases of obligations, we summarize the following features that is relevant for obligations. _ Generally, obligations are associated with some action re- 1In the GLB act, a consumer is an individual who obtains or has obtained a financial product or service from a financial institution for personal, family or household reasons. A customer is a consumer with a continuing relationship with a financial institution. quest2, i.e., a subject promises to fulfill some obligations sometime in order to perform a specific action on some objects now. There are cases in which specific obligations are only associated with some special objects in the policies without reference to an action. However, a corresponding action can still be identified practically because usually the action making these objects special is the action causing these obligations. _ Obligations have usually some specific temporal constraints. Some obligations should be fulfilled before an access is allowed and the result from the obligation fulfillment may affect the decision about an action request. We call this kind of obligations pre-obligations. Other obligations should be fulfilled after the action in the action request is performed. We call this kind of obligations post-obligations. Intuitively, there should be some time interval allocated for each obligation. Otherwise, a policy enforcement engine does not know when it can start evaluating policy conditions, and subjects in a post-obligation can legally avoid obligations by simply saying “I will do it in the future”. Some policies may require obligations to be fulfilled repeatedly until some condition becomes true. _ A subject’s obligation may result from another subject’s action, i.e., the subject of an obligation may be different from the subject who caused the obligation. For instance, when an operator discloses some children information to third parties, third parties may be required to fulfill similar obligations the operator has to fulfill. In some situations, the subject of an obligation may be the system itself, e.g., log access history. _ Some obligations may be conditional, that is, conditional obligations are only required to be fulfilled if some related condition becomes true. For instance, COPPA says that “An operator is required to send a new notice and request for consent to parents if there are material changes in the collection, use or disclosure practices to which the parent had previously agreed.”. Here, the material changes are the conditions that trigger the execution of the obligations “send a new notice” and “request for consent”.

D. Obligation Model for Privacy Policies

In this section, we present a formal obligation model for privacy policies that encompasses the features we discussed in the previous section. The model serves as the theoretical foundation for our later discussion on the analysis of obligations. Since obligations are actions that some subjects have to fulfill during some time interval, the obligation model introduces a temporal constraint component that clearly specifies such a time interval. As mentioned in the previous section, we are striving for a simple yet flexible mechanism to specify these temporal constraints. Two important requirements for such mechanism are that it should be able to support the specification of the most common temporal constraints and an efficient analysis of these constraints. As we mentioned in the previous section, the initiator of an obligation may differ from the user who causes the obligation; therefore a component used to indicate the initiator of an obligation is also added to the permission. Such a component makes it possible to identify a subject for an obligation that could be different from 2In the access control literature, the term “access request” is usually used instead of “action request”. However, in privacy policy, actions like “collect” and “disclose” are not an “access”, therefore, we use a more appropriate term “action request” to replace “access request” thereafter in this paper. 136the subject in the P-RBAC permission to which the obligation applies. This should not be interpreted as giving a blank permission to the subject of the obligation to execute the action imposed by the obligation. Independently of the obligation, the obligation subject should have a permission to execute the action in the time interval which the obligation must be fulfilled; otherwise the obligation will be violated. It is quite common that when defining a permission assignment, the subject of the obligation is not fully identified. In some cases it is assumed that the user that submitted the action request is the subject of the obligation. There are other cases in which the subject is expected to be from a set of users assigned to a particular role. In those cases the P-RBAC permission explicitly identifies the subject of the obligation using a special set of context variables listed in Table 1. The set of those special context variables is a subset of set CV of context variables of LC0. These variables are mainly used in conditions and as subjects of obligations. Their use avoids the introduction of new notation in the model to identify obligation subjects. Details of how they are used will be apparent after we introduce the model. Our temporal constraint model is based on a simple notion of time domain, that is, the pair (Z;_). In our context, each element of Z is referred to as a time instant and _ is a total order on Z. In what follows, given t; t0 2 Z, [t; t0] denotes the time interval starting at time instant t and ending at time instant t0. Next definition introduces a terse yet flexible definition for temporal constraints which is the key notion in our temporal constraint model.

Definition 2. A temporal constraint tc is a tuple (ts; te;

VIII. POLICY ANALYSIS

Large scale environments, such as enterprises, usually have to comply with complex access control policies and privacy policies. The more complex these policies are, the higher is the possibility that policies contain mistakes. Such situation can arise because of new requirements, new regulations, or just human mistakes. Therefore, there is a need for techniques to detect incorrect policies before they are deployed. For convenience in defining concepts in this section, we refer to an action together with the objects to which the action applies as an action pair.

A. Invalid Detection

In the proposed obligation model, the execution of an obligation can trigger the execution of another obligation. We refer to such phenomenon as obligation cascading. A user that performs an obligation also needs a permission. The permission may require the execution of some other obligations. The new obligations, in turn, may require the execution of more obligations. We refer to the action pairs involved in the obligation cascading for a permission as the cascading bag 5 of the permission.

B. Dominance of Obligations

In Core P-RBAC, given an action request, pre-obligations in all permissions that contain the action pair in the request have to be fulfilled before evaluating the conditions, and post-obligations in all applicable permissions have to be fulfilled in order to perform the action. Therefore, we can expect that some action request could lead to a large number of obligations returned, especially from illwritten policies. Therefore reducing the number of obligations to be executed may have significant practical impact. Obviously, the remaining obligations should not decrease the duty required by the original policies. On the other hand, we can imagine that many of these obligations are similar to each other since they are obligations associated with similar permissions. If the similarity can lead to some obligation relation like set containment, we may safely remove some obligations. In order to better understand the possibility before entering into details, we first discuss one example. Given two post-obligations, one requiring to send a privacy notice to both children and a parent within one week, and another requiring to send the same privacy notice to the parent within two weeks, if both of them are in the post-obligation set returned upon a user action request, it is reasonable that the user only needs to fulfill the former one because the duty represented in the latter one is “dominated” by the former one. In this paper, we use the term dominance to represent this relation, that is, the former obligation dominates the latter one. There are several factors affecting the dominance relation of obligations, and the first of them to be investigated is the temporal constraint.

C. Mapping Ownership

There is a simple way to model ownership by collapsing the owner roles into a set and only displaying the owner set in the role graph. This simplifies the display of the role graph. We can use a separate graph to manage ownership. Then RBACc = |R| n×m ≤ |Rregular |+|Radmin |+n n×m , since every object has at least one owner. This approach only simplifies the display; the number of roles in the system is unchanged, and the RBAC complexity is not changed. A second solution is moving the owner roles to a separate partition in the role graph, giving the role set three partitions: regular roles, owner roles and administrative roles. This approach also simplifies the display, but it does not reduce the total number of roles. The best solution is to use parameterized roles. With a parameterized owner role, when an owner activates the owner role, only the owner privileges relating to the objects that the user owns will be available. Since all the owner roles have similar “full control” privileges, these can be parameterized appropriately. Both owners and objects (in the access control sense) can be represented by objects with attributes, such as the userID for the user and the ownerID for the objects. Then constraints can be expressed so that the user’s userID must match the object’s ownerID for any ownership privileges assigned to the owner role. This way, we require only one owner role in the administrative part of the role graph, which greatly reduces the complexity. For the company with 40,000 employees, 40,000 owner roles can be reduced to one role. The RBAC complexity is RBACc =|R|n×m≤ |Rregular|+|Radmin|+1n×m or|Rregular|+|Radmin|n×m .

D. Mapping Granted Privileges

To model granting of regular privileges, we will look at the mechanism available in relational databases. In Oracle, the information is stored in a directory table, where privileges are marked as grantable and non-grantable. To model this in a role graph, we can create two sets of privileges accordingly: regular privileges Pr that cannot be granted further and grantable privileges Pg giving the set P = Pr SPg and |P| = 2|Pdb|, where Pdb is the set of database privileges that has all the access rights without the grantable mark. As well as the owner (parameterized) role above, we need an administrative role for the grantor. The owner has userrole assignment, create subrole and ownership transfer privileges. The grantor role also has the user-role assignment and create subrole privileges but does not have the ownership transfer privilege. The create subrole privilege is really a create role privilege, with the administrative domain just being the role containing the grantable privileges. In Figure 4, t represents a table, and we use bold to show grantable privileges. Except for DBUser which has only regular privileges, we do not show the regular privileges to save space. The two administrative roles are on the right side of the role plane. On the left side of the figure we show some of the nodes in the the user/group plane. The edge from group AssistantMgr to MarketingMgr can be used to indicate that the MarketingMgr has granted all of their privileges to the AssistantMgr. If the MarketingMgr is assigned to other roles (not shown), and only wishes to grant all privileges of MarketingDBManager, they can create a new group, say MarketingMgrGrantee, and make the AssistantMgr group a subgroup of this new group. If a user with grantable privileges wishes to grant only some of these privileges to another user, they can create a subrole with the privileges to be granted and assign this subrole to the grantee. The construction below makes sure that the grantor has all the required privileges to carry this out. This idea has two limitations: first, it is difficult to see who the grantor is; thus it is difficult to construct the granting path. Second, if the granting is total, then it is shown in the user/group plane on the left hand side of the graph; it is difficult to distinguish this from regular user group memberships. For example, the assistant manager is the grantee of the marketing manager, but in the user/group graph this is also a group membership, so it is difficult to show the difference between the granting relationship and group membership in this case. To solve the first problem, we can create a "helper" group and give it a name related to the grantor. For example, if the data entry clerk wants to grant the grantable insert privilege to a trainee, they can create a trainee role and put the grantable insert privilege into this role, and then create a group named DataEntryGrantor and assign the trainee users to this group. By doing this, we can see the grantor and grantee relationship clearly. This example can be seen in Figure 5. solve the second problem, we can change the name of the grantee to let the grantor-grantee relationship stand out. For example, we can change the name of AssistantMgr group to AssistantManagerGrantee; then we know that it is not group membership but rather a grantor-grantee relationship.

E. Analysis of the Mapping

Compared to the construction in Sandhu et al. [10], this model creates fewer roles. In their model, four roles are created for each privilege (i.e. for each object, access mode pair). Our model initially creates two administrative roles, plus a number of roles bound above by the number of users, and then creates additional roles or groups proportional to the number of grant operations. The subrole is added when necessary, whereas Sandhu's approach adds roles directly to the system when the object is created. We use both the group plane and the role plane, and add two administrative roles. Their approach adds three administrative roles per privilege, whether or not any granting is done. The number of roles generated by grant operations may be altered by revoke operations. We suspect that grant is far more common than revoke. We can argue that the roles created by granting operations represent useful sets of privileges or why else would they be involved in a grant operation. In any case, the RBAC complexity is incomparable with the previous work, but we feel that the extra roles and groups added to the graph are there because they are being used, and not just by a universal construction.

F. Expressive Power

An interesting issue is the expressive power of this access control model. In the simplest access control model conceivable, constraints are expressed purely in terms of subject sets and permission sets. For example, constraints are expressed by listing the set of permissions that each subject may be assigned (i.e., expression using propositional logic). This model has the nice feature that it is fail-safe in that only permissions that are allowed to be assigned to a subject may be. On the other hand, constraint specification is tedious and dynamic creation of objects requires the creation of new constraints before any rights may be assigned. On the other hand, a fully general model enables the use of universal and existential quantification on an arbitrary number of variables. Also, the full power of predicate logic would be available. The graphical constraint model (i.e., the constraint expression subset of the graphical access control model) contains the ability to express universal quantification over two sets, and provides set operations for use on the result. The only predicate in the model is precond, and even this has been added with some trepidation. A question is how close the expressive power of the graphic constraint model is to what is necessary in practice. Unfortunately, this is difficult to prove analytically. As one test, we show empirically in this paper the variety of practical constraints that can be expressed using the model. Second, we compare the expressive power of the graphical constraint model to another recent proposal for a practical constraint model, RSL 99 [Ahn and Sandhu 1999]. RSL99 supports a restricted first-order predicate logic in which a universal quantification over a predefined set of functions can be specified. The functions in RSL99 correspond to those in the graphical constraint model, except that the different dimensions of the functions, such as history, are not included. Thus,we expect that constraints based on these other dimensions will be more complex to specify. Clearly, RSL99 can express n-ary constraints, whereas the graphical constraint model can only specify binary constraints. However, at this point, we see most practical constraints as comparisons of two concepts: one set has a constraint in relation to another. The addition of ternary and greater concepts makes the language more complex, and we have reservations about system administrators' abilities to express higher-order constraints. RSL99 also includes typical set operations to create the sets used in the constraint comparison. We define aggregation and inheritance relations to the union operator to sets.We did not define an intersection operator for the graphical constraint model, as none of the example constraints warranted it. However, it can be added in a straightforward way.

RSL99 uses general, mathematical expressions for performing set comparisons. Therefore, arbitrary set comparisons can be made. In the graphical constraint model, we define a set of higher-level comparators that represent common mathematical expressions that are relevant to the example constraint types. For example, the disjoint comparator expresses a null intersection. We believe that maintaining a small set of intuitive comparators that cover the range of useful constraints will be key to model simplicity. RSL99 also includes operators to ease the expression of universal and existential quantification oneelement and allother, which take one element from a set and repeatedly extract all others, respectively. Our selection functions for sets and elements enable the same information to be expressed. In addition, we have an additional concept that is the iteration over the sets in an aggregation which we found useful in a number of cases. Ultimately, a comparison of constraint expression between the graphical access control model and RSL99 will require some empirical analysis of how different useful constraints are specified. Example 12 is the expression of a constraint that is also expressed in RSL99 [Ahn and Sandhu 1999]. The RSL99 expression for this constraint is j roles¤(OE(U)) \OE(CR) j ·1. The constraint expression in the graphical access control model is also somewhat complex (see Figure 15). However, the graphical representation reduces some expression complexity by defining the jj comparator rather than requiring the full intersection expression in RSL99. Also, the graphical representation of the sets involved in the constraint eliminates the need to express that part of the constraint in the language. Thus, the same expression in RSL99 is much shorter in our model. Quantification is still somewhat complex in both expressions, and we feel that more work is still necessary to make quantification manageable. In addition, the graphical model has another potential advantage that we are just beginning to leverage. Unlike a rule, the concepts in a graphical constraint have well defined semantics in the model, so a variety of analysis are possible. For example, we use the graphical access control model to estimate the complexity of safety verification using constraints [Jaeger 2001]. Further analyses, such as the identification of redundant or conflicting constraints,may also be useful. In summary, we think that the graphical constraint model and RSL99 share a great deal of common semantics about expressing access control constraints. The main differences are: (1) that the graphical access control model separates the steps of identifying the sets for comparison, selection the comparison inputs, and performing the comparison and (2) in the trade-off between expressive power and complexity for concepts such as quantification and set comparison and the means of expression of the constraints. In the graphical access control model, these steps in a constraint comparison are more explicit separate. The separate step of set identification is made straightforward in the graphical access control model, in particular. RSL99 provides more flexibility at a cost of additional complexity. We found this additional complexity unnecessary for our example constraints and are striving to keep the number of useful comparators small. We believe that more complex relations are beyond practical application, and the use of general mathematical expressions for set comparison deter from an administrator's ability to intuit the mean of a constraint.

G. Safety Verification

Safety verification involves computing all the constraints to determine if the comparisons are satisfied or not. The computation of a constraint involves the three steps made explicit by our constraint definition.

—Identification. First, the sets in the constraint are identified. This task involves using the functional definitions of the two sets in the binary relation to compute their membership. Such a computation may be optimized by effective caching of intermediate results.

—Selection. Next, the selection function determines how many comparisons will be necessary in order to verify the constraint and what those comparisons will be.
— Comparison. Lastly, perform the comparison(s) on the selected inputs.

The comparator function is executed on the inputs selected by the selection function. The computational complexity of safety verification per constraint is the sum of the identification complexity and the product of selection cost and comparison cost.

H. Computational Complexity

Any constraint model should enhance the performance of computing constraints. Again examining a trivial constraint model, the worst-case computation time to verify a safe configuration is O(jSjj Pj2). For each subject, we must determine whether their permissions are in the set of legal permissions. In Table I, we list the worst case computation time of each of the example constraints. The only constraint that is, in worst case, as expensive as the simple constraint model is Example 13, because this is a constraint between subjects and permissions as in the simple model. Constraint 11 is also interesting in that it is computed on the set of sessions as expressed in the worst-case analysis, which is larger than the set of subjects. However, the actual enforcement will be done per session (i.e., one session at a time). When a type is added to a session, it must be verified that the session types is not a superset of the process types (O(T2)). We briefly examine the computational complexity of the other examples. Examples 1 through 5 are simply intersection computations on a particular pair of sets. Therefore, they are all O(n2) where n is the size of the particular sets. Example 6 simply examines the objects that have been accessed by the subject to determine if a certain one has already been accessed. Example 7 may require that the unique type assignments must be collected (i.e., unioned) for each user, so its worst-case runtime is dominated by this. Example 8 requires two successive intersections (T(u1) \ T(u2) \ Trt), but the runtime of each intersection is the same. Examples 9, 10, and 12 may require the collection of types for each user, so their runtime is the same as Example 7. Example 14 may require that we collect all the operations that the user has run, and intersection them with the precondition operation set. One potential advantage of a graphical model is that we can direct the caching of the identified sets. Note that the selection and comparison complexities always are greater than that of identification, so the benefit of caching is limited to the constant factor of the computation. There are two problems that we must address in caching data: (1) determining what data is a candidate for caching and (2) determining whether caching that data provides a benefit. In the first case, as we see in Example 7, maintaining the set of types of all subjects in an aggregation at the aggregations S1 and S2 reduces the need to gather this information from each subject (i.e., reducing the selection cost to O(1)). Since there is a constraint on types on this aggregation, this indicates that maintaining the value of this function locally at the aggregation may improve the performance of the constraint check. On the other hand, maintaining the consistency of a set of distributed caches can be expensive as well. Therefore, it does not make sense to cache data that changes much more frequently than the configuration itself, such as activated permissions (e.g., Example 15).

I. Role Schedule Strategy

Under this strategy, we decompose the TRBAC analysis problem into multiple subproblems using schedules associated with the roles (srole). Let T (_,M) be a subproblem for time slot _ ∈ (0, TMAX), where a rule m ∈M if _ ⊆ sm role. Informally, a subproblem for a time slot contains all of the rules that is valid w.r.t its role schedule (i.e.: the rules that is authorized to change ER and TUR relations for that particular time slot). The details of this decomposition is explained in Section 6.

J. Rule Schedule Strategy

As noted earlier, in this strategy, we split the TRBAC analysis problem into multiple RBAC analysis problems using the rule schedules. In order to handle the RBAC problems, we have adapted the ideas from Stoller et al. [15] and modified them to suit to the temporal case. In the analysis, we keep track of different configurations c that can be reachable from an initial state c0. Since we only consider t can assign and t can revoke rules and one target user, c is composed of ([TUR) where [TUR ⊆ R × S.Hence in each configuration, we track (role, srole) pairs. In the algorithm, we trace constant regions C1, C2, ... serially with respect to time. These regions can be seen as separate RBAC systems. However, Ci+1 depends on Ci, ∀i, which implies the output of an RBAC reachability analysis at Ci is an input (or initial configuration) to Ci+1. Since an RBAC analysis could result in multiple configurations, then, in each constant region, a separate RBAC analysis should be performed for each configuration generated by the analysis done in the previous constant region. Moreover, there are other issues related to role schedules that are assigned by the rules. Recall that all of the rules have a role schedule which denotes the time intervals that the role can be assigned. But, according to the rule definitions, the administrators are free to choose a sub schedule of the role schedule and assign/revoke the role only for some of the designated time intervals. This further complicates the reachability analysis, since in a serial fashion, one should keep all of the possible schedule combinations for the subsequent time intervals. Therefore we make the following assumption to simplify the algorithm: Sub-schedule Assumption: For each t can assign and t can revoke rule, the assignment and revocation operations are performed using the entire schedule srole. In other words, assume that a schedule srole covers all time slots. This means that an administrator may use this rule to assign the associated role r to a user u, all of the subsets of the schedule srole (as long as the preconditions are satisfied). In our analysis, we assume that srole is assigned or revoked completely - no sub schedule assignments are allowed. Hence, this assumption ensures that a rule can generate only one (new) configuration, which is actually similar to the non temporal analysis. Here we provide a sketch of the algorithm. The TRBAC reachability analysis starts with an initial configuration c0 and constant region C1. The state space is expanded using Stoller's algorithm and the rules that are valid at time t = 0. At the end of this step, a set of reachable configurations, S1 = {c1, c2, ..., cM} are obtained. Afterwards, the analysis moves to C2. For each distinct configuration obtained so far, Stoller's algorithm is used to expand these configurations using the valid rules in this constant region. At the end of this step, we obtain an updated set of reachable configurations S2 ⊇ S1. The algorithm then moves to C3 and the trace goes in this fashion for a specified number of cycles P of length TMAX (The algorithm returns to C1 whenever TMAX is reached). Since TURA tuple ST is finite and since the iterations are bounded by the number of cycles, the algorithm is guaranteed to terminate. However since this approach is a greedy heuristic, we are not guaranteed to get an optimal

IX. AUTHENTICATION USING RFID

In order to enforce visibility policies the player must be able to reliably compute the predicate σ(Xi,L(Oj)) of an object's trajectory and supply it as attributes in the access request to the PDP. This is challenging, since the necessary information about the trajectory is distributed across multiple players and need not be known to the player. Each player knows by default only his predecessor and successor from physically moving the object which is even insufficient to determine its own rank in the trajectory. Therefore the requestor must supply the predicate, but of course the information is unreliable, since he should not necessarily be trusted. The player must verify the supplied predicate. This problem is an extension of the authentication problem in distributed systems. Clearly authentication is a prerequisite for any access control, but as seen before our predicate is an attribute of subject and resource, and as such common identity verification mechanisms fall short of solving the problem. Nevertheless, similar to the most common solutions for the authentication problem in distributed problems, we can resort to cryptographic techniques. We are concerned about physical objects (equipped with an RFID tag each) and a player needs to prove possession of this object. Differently from the ownership authentication factors – "something you have" – our authentication must succeed even if the player is no longer in possession of the object – "something you had". This complicates the problem, since it rules out solutions of simply interactively using RFID for access control [7, 41]. The notion of (current) possession has been explored before and extended to securely verifying ownership of an RFID tag [34, 42]. This concept already has many applications for mobile physical objects in supply chains, but, e.g. for any form of analytics, authentication and possession are decoupled. Also as pointed out in [13] these protocols still suffer from security flaws. The problem of authenticating based on (past) possession of RFID tags has been first considered in [30]. Yet these protocols do not allow implementing our predicate, but simply allow a decision whether an item has been in possession. They therefore do not allow a distinction between upstream and downstream. Our authentication relies on a similar mechanism as the proofs of possession from [18]. A proof of possession is in our terms a verifiable predicate which can be supplied during the access request. Unfortunately all solutions proposed in [18] are either not reliable in our attacker model or are not realizable in our system model. A different design is therefore needed.

A. System Model

We continue our model with multiple players, but restrict ourselves in this section to one object Oj which is the one considered in the access request. One player (Xv) is the designated verifier of the predicate. We assume each player is uniquely identifiable and the availability of a public-key infrastructure to securely distribute public keys for each player. Besides the basic capabilities for communication, we only assume the availability of re-writeable permanent storage on the RFID tag. Passive tags (without own source of power) with up to 64 KBytes of storage are available [17] and follow the EPC Gen 2 protocol which is commonly used in supply chains [1]. Note that we do not consider cryptographic capabilities on the RFID tag, such as symmetric encryption [15] or publickey cryptography [5, 8, 21]. We emphasize that is a very strong restriction of the solution space. It implies that the RFID tag cannot manage secret material, such as cryptographic keys or even passwords. It therefore rules out any solution transferring common concepts from distributed systems authentication. E.g., signing challenges by the RFID tag using message authentication codes or public-key signatures which significantly simplify the problem cannot be implemented in our model. We do this in order to address the security problems of existing and currently emerging deployments of RFID in supply chains which do not yet have these cryptographic capabilities. Before the access request and the problem of authentication, several operations are performed using the RFID tag. We use a simplified model from [30]. Assume Trent (T) is a trusted third party that supports players in obtaining RFID tags. A natural choice is the RFID manufacturer. Note that Trent does not obtain any additional information about the supply chain operation than any RFID manufacturer already does now. Our authentication consists of the following algorithms or protocols. Initialize: A player Alice requests a (or a set of) RFID tags from Trent. She can later use those to attach to newly created objects. Move: A player Alice moves an object to another player Bob. She updates the information stored on the attached RFID tag. We emphasize that this operation does not require network access to Trent or Bob. Authenticate: The requestor sends a verifiable predicate σ(Xi,L(Oj)) for access to the verifier. The verifier makes an access control decision based on this predicate (and its policies).

B. Security Model

We assume secure and authenticated communication over the network, i.e. between the players and Trent. We assume insecure communication with the RFID tag attached to the object. Our attacker controls the requestor (Xi) and may control any other player except the verifier (Xv) and Trent, i.e. we consider almost arbitrary collusion. Our attacker is adaptive, i.e. the set of controlled players may change over time.

Differing from attacks targeting network availability, attacks targeting data integrity can be regarded as less brute-force yet more sophisticated attacks. The target of the attacks is either customer's information (e.g., pricing information and customer account balance) or network operation information (e.g., voltage readings, device running status). In other words, such attacks attempt to deliberately modify information shared within the smart grid in order to corrupt critical data exchange in the smart grid. On the contrary, attackers targeting information privacy do not attempt to modify information transmitted over power networks but to eavesdrop on communications in power networks to acquire desired information, such as a customer's account number and electricity usage. Such attacks can be considered to have negligible effect on the functionality of the communication networks in the smart grid. Consequently, compared with attacks targeting data integrity, attacks targeting information privacy may not lead to catastrophic consequences, such as massive blackout. The risk of attacks targeting data integrity in the power networks is indeed real. A notable example is the recent work of [18], which proposed a new type of attacks, called false data injection attacks, against the state estimation in the power grid. The paper assumed that an attacker has already compromised one or several meters and pointed out that the attacker can take advantage of the configuration of a power system to launch attacks by injecting false data to the monitoring center, which can legitimately pass the data integrity check used in current power systems. More recently, new methods [19] have been developed to provide state estimation that is robust to the false data injection attacks. In order to launch attacks that attempt to compromise data integrity or to acquire privacy information, an attacker has to first stealthily intrude the computer system of a legitimate node, or by some means access a power network with authentication. Therefore, the design of countermeasures to attacks targeting data integrity and information privacy can consist of the following perspectives.

C. Authentication Protocol Design

Authentication is an important identification problem for any communication network. Strong authentication schemes are required for customers and electronic devices to ensure communications with full security and to meet the stringent requirements of the communication network in the smart grid, such as message delay and power consumption constraints. To this end, existing work [20]–[22] in general aims at providing efficient and fast authentication protocols for a variety of power subsystems, including transmission and operation systems, distribution networks, and customers' home-area networks. For example, the work of [22] showed that the time critical constraint implicitly results in the following requirements for the design of authentication protocols:

(i) Efficient algorithms to minimize computational cost,
(ii) Low communication overhead, and
(iii) Robustness to attacks

Towards these goals, the work in [22] and [23] focused on the design of authentication protocols to meet the requirements for the low latency and DoS attack resilience.

D. Intrusion Detection

The smart grid must have the ability to detect the attempt of an intruder to gain unauthorized access to computer systems. Recently, a few papers have investigated the problem of cyber intrusion detection in power networks [24]–[26]. In general, the intrusion detection for computer systems falls mainly into the cyber security field and has been well studied in the literature.

E. Firewall and Gateway Design

As mentioned before, differing from the Internet, the smart grid has only two major directional information flows: bottom-up and top down. Thus, it will be easy for gateway or firewall software's to perform traffic control on information flows in smart grid to block undesired or even suspicious flows generated by malicious nodes. Note that it may be non-trivial to assume an attacker can easily compromise a legitimate node or access the power network with authentication. But due to the ubiquitousness of the smart grid network, it is still possible that a malicious attacker can, by some means, connect to a power network and launch attacks targeting data integrity or information privacy.

F. Authors and Affiliations

Dr Akash Singh is working with IBM Corporation as an IT Architect and has been designing Mission Critical System and Service Solutions; He has published papers in IEEE and other International Conferences and Journals.

He joined IBM in Jul 2003 as a IT Architect which conducts research and design of High Performance Smart Grid Services and Systems and design mission critical architecture for High Performance Computing Platform and Computational Intelligence and High Speed Communication systems. He is a member of IEEE (Institute for Electrical and Electronics Engineers), the AAAI (Association for the Advancement of Artificial Intelligence) and the AACR (American Association for Cancer Research). He is the recipient of numerous awards from World Congress in Computer Science, Computer Engineering and Applied Computing 2010, 2011, and IP Multimedia System 2008 and Billing and Roaming 2008. He is active research in the field of Artificial Intelligence and advancement in Medical Systems. He is in Industry for 18 Years where he performed various role to provide the Leadership in Information Technology and Cutting edge Technology.

REFERENCES

[1] Dynamics and Control of Large Electric Power Systems. Ilic, M. and Zaborszky, J. John Wiley & Sons, Inc. © 2000, p. 756.

[2] Modeling and Evaluation of Intrusion Tolerant Systems Based on Dynamic Diversity Backups. Meng, K. et al. Proceedings of the 2009 International Symposium on Information Processing (ISIP'09). Huangshan, P. R. China, August 21-23, 2009, pp. 101–104

[3] Characterizing Intrusion Tolerant Systems Using A State Transition Model. Gong, F. et al., April 24, 2010.

[4] Energy Assurance Daily, September 27, 2007. U.S. Department of Energy, Office of Electricity Delivery and Energy Reliability, Infrastructure Security and Energy Restoration Division. April 25, 2010.

[5] CENTIBOTS Large Scale Robot Teams. Konoledge, Kurt et al. Artificial Intelligence Center, SRI International, Menlo Park, CA 2003.

[6] Handling Communication Restrictions and Team Formation in Congestion Games, Agogino, A. and Tumer, K. Journal of Autonomous Agents and Multi Agent Systems, 13(1):97–115, 2006.

[7] Robotics and Autonomous Systems Research, School of Mechanical, Industrial and Manufacturing Engineering, College of Engineering, Oregon State University

[8] D. Dietrich, D. Bruckner, G. Zucker, and P. Palensky, "Communication and computation in buildings: A short introduction and overview," *IEEE Trans. Ind. Electron.*, vol. 57, no. 11, pp. 3577–3584, Nov. 2010.

[9] V. C. Gungor and F. C. Lambert, "A survey on communication networks for electric system automation," *Comput. Networks*, vol. 50, pp. 877–897, May 2006.

[10] S. Paudyal, C. Canizares, and K. Bhattacharya, "Optimal operation of distribution feeders in smart grids," *IEEE Trans. Ind. Electron.*, vol. 58, no. 10, pp. 4495–4503, Oct. 2011.

[11] D. M. Laverty, D. J. Morrow, R. Best, and P. A. Crossley, "Telecommunications for smart grid: Backhaul solutions for the distribution network," in *Proc. IEEE Power and Energy Society General Meeting*, Jul. 25–29, 2010, pp. 1–6.

[12] L. Wenpeng, D. Sharp, and S. Lancashire, "Smart grid communication network capacity planning for power utilities," in *Proc. IEEE PES, Transmission Distrib. Conf. Expo.*, Apr. 19–22, 2010, pp. 1–4.

[13] Y. Peizhong, A. Iwayemi, and C. Zhou, "Developing ZigBee deployment guideline under WiFi interference for smart grid applications," *IEEE Trans. Smart Grid*, vol. 2, no. 1, pp. 110–120, Mar. 2011.

[14] C. Gezer and C. Buratti, "A ZigBee smart energy implementation for energy efficient buildings," in *Proc. IEEE 73rd Veh. Technol. Conf. (VTC Spring)*, May 15–18, 2011, pp. 1–5.

[15] R. P. Lewis, P. Igic, and Z. Zhongfu, "Assessment of communication methods for smart electricity metering in the U.K.," in *Proc. IEEE PES/IAS Conf. Sustainable Alternative Energy (SAE)*, Sep. 2009, pp. 1–4.

[16] A. Yarali, "Wireless mesh networking technology for commercial and industrial customers," in *Proc. Elect. Comput. Eng., CCECE*,May 1–4, 2008, pp. 000047–000052.

[17] M. Y. Zhai, "Transmission characteristics of low-voltage distribution networks in China under the smart grids environment," *IEEE Trans. Power Delivery*, vol. 26, no. 1, pp. 173–180, Jan. 2011.

[18] V. Paruchuri, A. Durresi, and M. Ramesh, "Securing powerline communications," in *Proc. IEEE Int. Symp. Power Line Commun. Appl., (ISPLC)*, Apr. 2–4, 2008, pp. 64–69.

[19] Q.Yang, J. A. Barria, and T. C. Green, "Communication infrastructures for distributed control of power distribution networks," *IEEE Trans. Ind. Inform.*, vol. 7, no. 2, pp. 316–327, May 2011.

[20] T. Sauter and M. Lobashov, "End-to-end communication architecture for smart grids," *IEEE Trans. Ind. Electron.*, vol. 58, no. 4, pp. 1218–1228, Apr. 2011.

[21] K. Moslehi and R. Kumar, "Smart grid—A reliability perspective," *Innovative Smart Grid Technologies (ISGT)*, pp. 1–8, Jan. 19–21, 2010.

[22] Southern Company Services, Inc., "Comments request for information on smart grid communications requirements," Jul. 2010

[23] R. Bo and F. Li, "Probabilistic LMP forecasting considering load uncertainty," *IEEE Trans. Power Syst.*, vol. 24, pp. 1279–1289, Aug. 2009.

[24] *Power Line Communications*, H. Ferreira, L. Lampe, J. Newbury, and T. Swart (Editors), Eds. New York: Wiley, 2010.

[25] G. Bumiller, "Single frequency network technology for fast ad hoc communication networks over power lines," WiKu-Wissenschaftsverlag Dr. Stein 2010.

[31] G. Bumiller, L. Lampe, and H. Hrasnica, "Power line communications for large-scale control and automation systems," *IEEE Commun. Mag.*, vol. 48, no. 4, pp. 106–113, Apr. 2010.

[32] M. Biagi and L. Lampe, "Location assisted routing techniques for power line communication in smart grids," in *Proc. IEEE Int. Conf. Smart Grid Commun.*, 2010, pp. 274–278.

[33] J. Sanchez, P. Ruiz, and R. Marin-Perez, "Beacon-less geographic routing made partical: Challenges, design guidelines and protocols," *IEEE Commun. Mag.*, vol. 47, no. 8, pp. 85–91, Aug. 2009.

[34] N. Bressan, L. Bazzaco, N. Bui, P. Casari, L. Vangelista, and M. Zorzi, "The deployment of a smart monitoring system using wireless sensors and actuators networks," in *Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, 2010, pp. 49–54.

[35] S. Dawson-Haggerty, A. Tavakoli, and D. Culler, "Hydro: A hybrid routing protocol for low-power and lossy networks," in *Proc. IEEE Int. Conf. Smart Grid Commun. (SmartGridComm)*, 2010, pp. 268–273.

[36] S. Goldfisher and S. J. Tanabe, "IEEE 1901 access system: An overview of its uniqueness and motivation," *IEEE Commun. Mag.*, vol. 48, no. 10, pp. 150–157, Oct. 2010.

[37] V. C. Gungor, D. Sahin, T. Kocak, and S. Ergüt, "Smart grid communications and networking," Türk Telekom, Tech. Rep. 11316-01, Apr 2011.