DIPS: A Framework for Distributed Intrusion Prediction and Prevention Using Hidden Markov Models and Online Fuzzy Risk Assessment

Kjetil Haslum, Ajith Abraham and Svein Knapskog
Center for Quantifiable Quality of Service in Communication Systems
Norwegian University of Science and Technology
O.S. Bragstads plass 2E, N-7491 Trondheim, Norway
haslum@q2s.ntnu.no, ajith.abraham@q2s.ntnu.no, knapskog@q2s.ntnu.no

Abstract

This paper proposes a Distributed Intrusion Prevention System (DIPS), which consists of several IPS over a large network (s), all of which communicate with each other or with a central server, that facilitates advanced network monitoring. A Hidden Markov Model is proposed for sensing intrusions in a distributed environment and to make a one step ahead prediction against possible serious intrusions. DIPS is activated based on the predicted threat level and risk assessment of the protected assets. Intrusions attempts are blocked based on (1) a serious attack that has already occurred (2) rate of packet flow (3) prediction of possible serious intrusions and (4) online risk assessment of the assets possibly available to the intruder. The focus of this paper is on the distributed monitoring of intrusion attempts, the one step ahead prediction of such attempts and online risk assessment using fuzzy inference systems. Preliminary experiment results indicate that the proposed framework is efficient for real time distributed intrusion monitoring and prevention.

1. Introduction

Firewalls are employed only at the network perimeter and they are not always effective against intrusion attempts. The average firewall is designed to filter detect and deny clearly suspicious traffic. Many attacks, intentional or otherwise, are launched from within an organisation. Intrusion detection systems may be effective at detecting suspicious activity, but do not provide protection against attacks. In Distributed IDS (DIDS), conventional intrusion detection system are embedded inside intelligent agents and are deployed over a network. In a distributed environment, IDS agents communicate with each other, or with a central server. By having these co-operative agents distributed across a network, incident analysts, network managers and security personnel are able to get a broader view of what is occurring on their network as a whole. Distributed monitoring allows early detection of planned and coordinated attacks, thereby allowing the network managers to take preventive measures. In a DIDS, it is important to ensure that the individual IDS is light-weight and accurate. In the DIPS framework, code fragments developed using genetic programming models are embedded inside intelligent agents (IDS) to detect various types of attacks [1]. Individual IDS sensor node outputs are provided as inputs to the Hidden Markov Model (HMM).

The rest of the paper is organised as follows. Section 2 introduces key concepts of distributed intrusion prevention systems and the technical requirements to design such systems in practice. Section 3 deals with HMM followed by some experimental results in Section 4. Online risk assessment by fuzzy inference system is presented in Section 5 and some conclusions are provided towards the end.

2. Intrusion Prevention Systems (IPS)

Intrusion prevention systems are proactive defence mechanisms designed to detect malicious packets within normal network traffic, block the offending traffic automatically before it does any damage. Like IDS, IPS may be also classified as Host based IPS or Network based IPS. There are a number of challenges to the implementation of an IPS device in addition to those to be faced when deploying passive-mode IDS products. These challenges all stem from the fact that the IPS device is designed to work inline, presenting a potential choke point and single point of failure. Some of these problems could be eliminated in a distributed intrusion prevention system, where there is no single point of control and the problems are tackled at its source of origin as much as possible. The main task of the IPS is to block a suspect traffic flow as soon as possible by

0-7695-2876-7/07 $25.00 © 2007 IEEE
DOI 10.1109/IAS.2007.67

immediately discarding suspect information packets. The suspicious traffic may also be re-routed for further forensic analysis etc. An IPS should have a maximum up time since it has the potential to close a vital network path and thus, inadvertently, cause a DoS condition. An IPS should be computationally light since it is essential that its impact on overall network performance is minimal and also achieve high packet processing rates. An IPS should minimize false positives since this can lead to a Denial of Service condition. IPS should be able to decide exactly which malicious traffic is blocked, provide a mechanism for alerts and have forensic analysis capabilities.

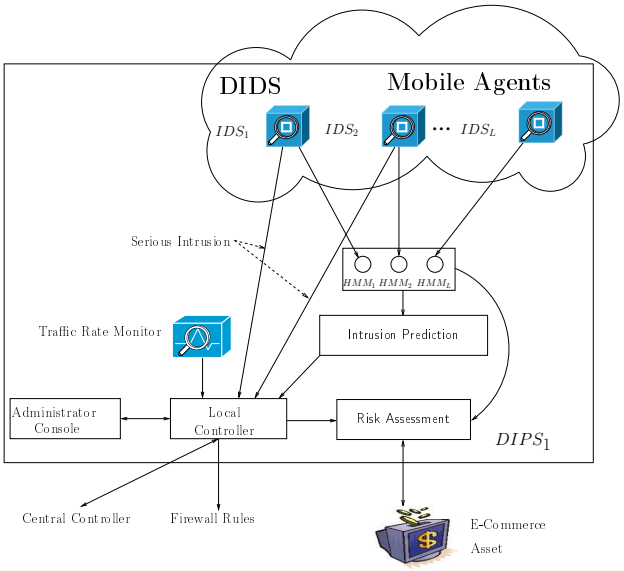

Figure 1. Architecture of a DIPS element.

2.1. Distributed Intrusion Prevention Systems (DIPS)

DIPS are simply a superset of the conventional IPS implemented in a distributed environment. We consider IPS as an integrated IDS with many more functions as listed in Section 2. Due to the distributed nature of IPS, the implementation poses several challenges. IDS are embedded inside software mobile agents and placed in the network to be monitored. The individual IDS may be configured to detect a single attack, or they may detect several types of attacks.

Figure 1 illustrates the basic architecture of a DIPS element, which is controlled by a local controller. In a large network, each DIPS element communicates/coordinates with other DIPS local controller and/or a central controller [2]. A HMM model processes the attack data information from the various mobile agent IDS sensors. IDS deployed are capable of detecting simple problems to serious denial of service type of attacks. Based on the nature of the detected attack, the following actions would be taken:

1. If the detected attack is simply a port scan or a probe, the HMM model will attempt to make a prediction of a possible future attack based on the current distributed attack pattern. Based on this prediction, the central controller (or administrator) would take precautionary measures to prevent future attacks. The central controller would also make use of an online risk assessment of the assets subjected to this possible serious attack in the future.

2. If the detected attack is very serious, the central controller would take necessary actions to re-configure firewall rules or notify the administrator etc. Such serious attacks would bypass the HMM model.

3. At any time any abnormal traffic rate is noted by the monitor, then again the central controller would take necessary actions to re-configure firewall rules or notify the administrator etc.

In the DIPS framework, each network component may host one or many IDS. Since there will be a large number of flag generators, these must be extracted, summarised, analyzed, and condensed by a suitable architecture before arriving at a final conclusion. Very often, it is to be noted that the event information, which is detected by the IDS agents will follow a bottom up approach for analysis and the various command and control flows will follow a top-down approach. The physical location of IDS agents may be fixed or mobile so as to monitor certain parts of the network segments.

The co-operative intelligent agent network is one of the most important components of the DIDS [2]. Ideally these agents will be located on separate network segments, and very often geographically separated. Communication among the agents is done utilizing TCP/IP sockets. Agent modules running on host machines are capable of data analysis and to formulate adequate response actions and are very often implemented as read only and fragile. In the event of tampering or modification the agent reports to the server agent and automatically ends its life. Agents residing in the individual analyzer/controllers consist of modules responsible for agent regeneration, dispatch, updating and maintaining intrusion signatures and so on. These agents control the individual IDS agents for monitoring the network, manage all the communication and life cycle of the IDS agents and also updates the IDS agents with detection algorithms, response and trace mechanisms

3. Hidden Markov Modeling of DIPS

Gao et al. [4] developed an HMM to predict attacks in the application layer and they claimed that the approach could be extended for network layer. Årnes et al. [3] used

HMMs for real time risk assessment, but not directly for attack prediction as we proposed in this paper.

A HMM can be described as two stochastic processes; the hidden process $(x_t; t = 1, 2, \ldots)$ that representing the state of the system, and the observable process $(y_t; t = 1, 2, \ldots)$ representing the observations made by an IDS Agent. There will be no direct relation between the t index and time. t will be a sequence number for observations received from the IDS agents. The HMM model used in this paper is described as follows:

- A set of states $S = \{s_1, s_2, \ldots, s_N\}$ describing the possible states of the system. To simplify the notation of equations and algorithms we will use i instead of s_i. In this paper only four states are used; *Normal(N)* indicating no suspicious activity, *Intrusion Attempt (IA)* indicating suspicious activity against the network, e.g. probing, *Intrusion in Progress (IP)* indicating that one or more attacker have started an attack against the system, and *Successful Attack (SA)* indicating that one or more attacker have broken into the system. The state space used in this paper is simlar to the statespace used in [6].

- A set of observations $V = \{v_1, v_2, \ldots, v_M\}$. To simplify the notation we will use k instead of v_k. For this paper, we assume, that each IDS Agent only produces three different types of observations; *No suspicious activity (N)*, *Probing (P)* indicating suspicious activity against the network, and *Successful Attack (SA)* indicating that an IDS Agent have detected a successful attack.

- An initial distribution vector $\pi = \{\pi_i\},\ \pi_i = P(x_1 = i)$ describing the state of the system when the monitoring starts. We assume the system to be in the N-state when monitoring starts.

- A transition probability matrix $P = \{p_{ij}\},\ p_{ij} = P(x_t = j | x_{t-1} = i)$, describing the dynamics of the interaction between the intruder and the system.

- An observation probability matrix for each of the L IDS Agents $Q_k^l = \{q_i^l(k)\},\ q_i^l(k) = P(y_t^l = k | x_t = i)$, describing the quality or the trustworthiness of each IDS Agent.

The HMM model used in this paper models only integrity and confidentiality, and make no attempts to model availability. We believe that availability is best modeled separately. The Markov Model used in this paper is shown in Figure 2. States drawn as circles indicate secure states, and state drawn by a square indicates that damage has already happened. When using a discrete HMM to model the system, we can make the following assumptions; all information about the system is contained in the state of the system, observations are independent given the current state, and state occupation times are geometrically distributed.

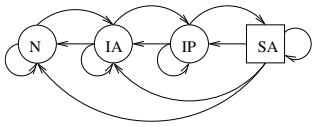

Figure 2. A Markov model modeling the security of a small network

Assume that we have a sequence of observations from each IDS agent $Y_t = \{Y_t^l\}_{l=1,\ldots,L}$, where $Y_t^l = y_1^l, \ldots, y_t^l$, and a model λ. For each of the L IDS Agents the probability of being in each of the N states is calculated only based on observations made by the corresponding IDS Agent $\gamma_t^l(i) = P(x_t = i | Y_t^l, \lambda)$. The computations required for updating the probability distribution γ_t^l can be found as Eq. 19 and Eq. 27 in [8].

The initial distribution π is used to initialize γ before the system starts to process observations from the IDS agents $\gamma_0^l = \pi, l = 1, \ldots, L$.

When an IDS Agent l detects suspicious activity in the network, it sends an observation y_t^l to the HMM, that updates γ_t^l.

After γ is updated the probability of being attacked (PA) is calculated based on the probability of being in the *IP* state. The PA can take on one of the three values; *Low*, *Medium* and *High*. A message with the current PA is sent to the Central Controller, to be presented for the administrator through the *Administrator Console* and for updating the *Intrusion frequency* as described in Section 5.

4. Experimental Results Using HMM

In order to demonstrate how HMMs can be used in DIPS, we have constructed a model of a small network as illustrated in Figure 3 and run some simulations to illustrate the proposed model. The example network consists of four different assets; a router, a public web server, a file server, and a database. Five IDS Agents denoted $IDS_1, \ldots, IDS_5$ are deployed in the network, and the observations are sent to the their corresponding HMM. We have generated a sequence of 32 observations for each of the five IDS agents $Y_1, \cdots, Y_5$. Figure 4 shows (from top to bottom); the hidden state X, observation sent from IDS Agent 1, and the probability of being in state Intrusion in Progress estimated by HMM 1 based on the observations from IDS Agent 1.

In the experiments, we have used four different states and three different observation symbols as described in Sec-

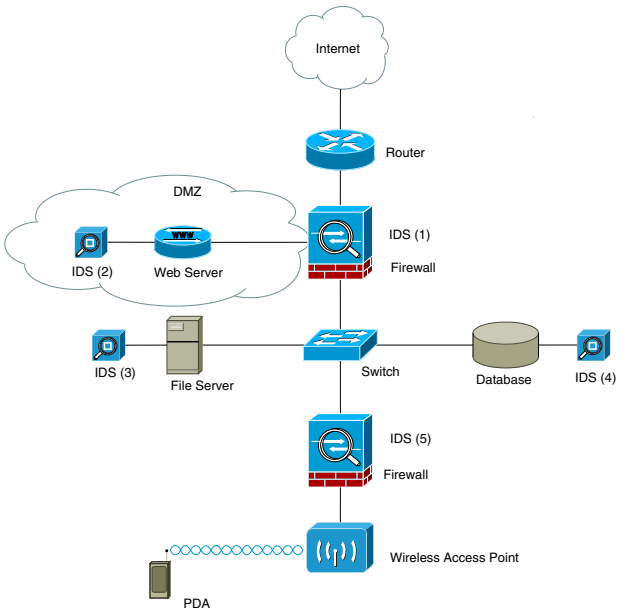

Figure 3. Example network showing assets and IDS agents

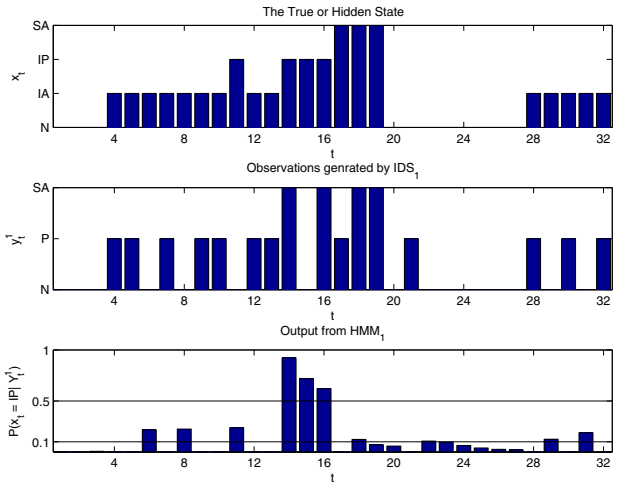

Figure 4. The hidden state, observations from the IDS and output from the HMM.

tion 3. This is illustrated in Figure 2. For this illustration, we also assume that all the IDS Agents send the observations at fixed time intervals to the corresponding HMM. Even if there are no attacks or suspicious activities, the N observations will be sent.

The system is assumed to be in the *Normal* state when the IPS starts. This corresponds to the following initial distribution $\pi = (1, 0, 0, 0)$. We have used a supervised training method to estimate the transition probability matrix P and the observation probability matrix Q. By supervised training, we mean that the hidden state X, is used in the estimation of P and Q. The P matrix is estimated by counting the number of transitions, and Q is estimated by counting the number of emitted symbols for each state. All observations from the five sensors were used to estimate one common Q, used in all the five HMMs. Events corresponds to the time when observations are received from the IDS agents. We assume that all observations are received at the same time, to make the figures more readable.

The upper graph in Figure 4 shows the hidden state used for the parameter estimation. The graph in the middle illustrates the output from the first IDS Agent, and the lower graph depicts the probability of being in the IP-state.

PA_l is estimated for each of the L IDS Agent based on the probability $\gamma_t^l(IP) = P(x_t = IP|y_t^l, \lambda_l)$ when the corresponding HMM is updated. The probability levels used to determine the PA_l is 0.1 and 0.5, also shown in Figure 4.

Figure 5 shows the AP from the five IDS Agents, which depicts how the output from the IPS may be visualized by the system administrator. It is observed that three of the five IDS agents reported higher risk of being attacked at $t = 16$, which is just one event before the first attack at $t = 17$.

High PA may lead to automated response like updating the fire wall rules or removal of users from this system. This kind of automated response should probably not always be based only on results from the HMM, but preferably also include some kind of automated forensic analysis based on traffic- and log-data stored in a database. High risk may trigger extended logging.

5. Modeling Risk Assessment Using Hierarchical Fuzzy Inference System

Risk analysis is fundamentally all about establishing probabilities. In the DIPS framework, we model the risk analysis using *threat levels*, *vulnerability* and *asset value* [5]. We consider that all components within a network scenario falls into one of these categories, and each has attributes, or derived factors, that contribute positively or negatively to risk.

Threat level is modeled as the frequency of attacks/intrusions, obtained from HMM predictions as described in Section 3, the probability that an intruder is being successful in overcoming protective controls and gains access to act against the organization or assets and the type and severity of attacks.

Vulnerability may be defined as the probability that an asset will be unable to resist the actions of an intruder.

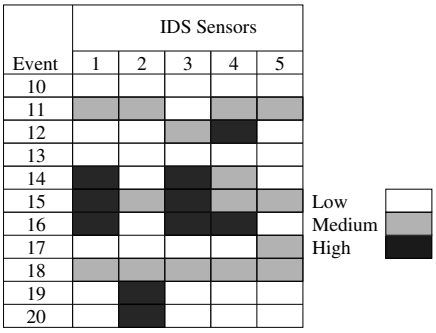

Figure 5. Estimated probability of being attacked, based on observations from the five IDS Agents.

Vulnerability exists when this probability exceeds a given threshold. This may be because of weaknesses in software or hardware, missing software patches and so on. Vulnerability may be modeled as contemporary high threat capability and low system threat resistance.

Asset may be defined as any data, device, or other component of the environment that supports information-related activities, and which can be affected in a manner that result in loss. To determine asset loss could be one of the hardest tasks of analyzing risk. It is very difficult to put a precise value on the various types of assets, and there may be more than one value or liability characteristic. Complex relationships might exist between the different forms of loss and many factors determine loss magnitude. We model asset value/loss as cost, criticality, sensitivity and recovery.

The overall architecture for asset risk management is summarised in Figure 6.

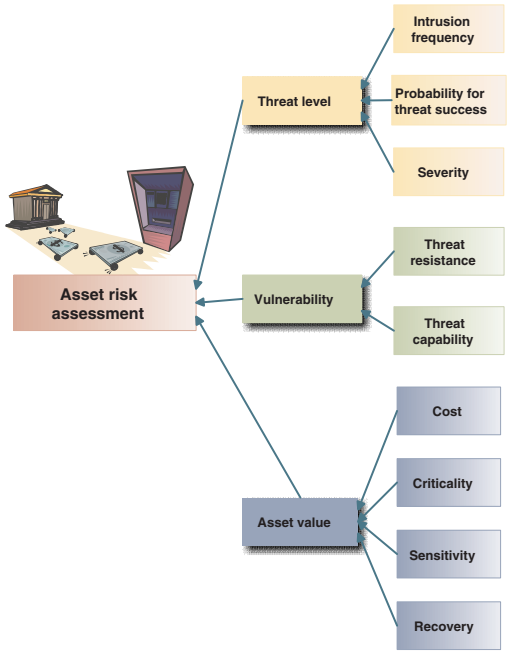

Figure 6. Generic structure of the risk assessment model.

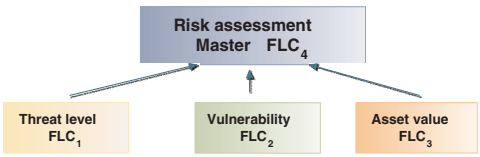

Figure 7. Hierarchical architecture of four fuzzy logic controllers.

5.1. Hierarchical Fuzzy Modeling of Risk Assessment

Most of the uncertainties in the risk assessment models are handled using statistical approaches. However, such methods cannot handle sources of imprecision that therefore may lead to uncertainty including scarce or incomplete data, measurement error, data obtained from expert judgment, or subjective interpretation of available information. In this paper we propose the use of fuzzy set theory to incorporate uncertainties into online risk assessment for DIPS. Based on the form of available information, fuzzy set theory, probability theory, or a combination of both can be used to incorporate the uncertainty and variability of risk variables into various risk assessment models.

Zadeh [9] introduced the concept of fuzzy logic to present vagueness in linguistics, and further implement and express human knowledge and inference capability in a natural way. We used an hierarchical fuzzy logic controller to asses the overall risk based on *threat level*, *vulnerability* and *asset value*. A fuzzy logic controller (FLC) is composed of a knowledge base, a fuzzification interface, an inference system and a defuzzification interface. The architecture of the hierarchical fuzzy logic controller for risk assessment is depicted in Figure 7. A FLC is assigned to make the inference from each of the three input variables *threat level*, *vulnerability* and *asset value* and a fourth FLC is assigned to make the overall inference for risk assessment. The Mamdani inference method [7] was used for all the four FLC's.

5.2. Fuzzy Modeling of Threat Level

Threat level is modeled as (1) frequency of attacks/intrusions (2) probability that intruder being success-

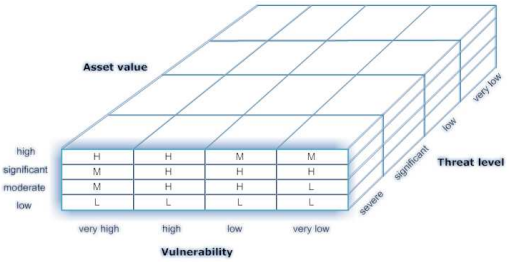

Figure 8. Fuzzy associative memory structure for FLC$_4$.

ful in overcoming protective controls and (3) Type and severity of attack. A Fuzzy logic controller (FLC$_1$) observes three input variables and produce one output variable. Three triangular membership functions are assigned per input variable and three triangular membership functions are used for the output variable.

5.3. Fuzzy Modeling of Vulnerability

Vulnerability is modeled as (1) threat capability and (2) system threat resistance. Three triangular membership functions are assigned per input variable and three triangular membership functions are used for the output variable. A Fuzzy logic controller (FLC$_2$) observes two input variables and produces the output variable.

5.4. Fuzzy Modeling of Asset Value and Loss

Asset value/loss is modelled using four variables: (1) cost (2) criticality (3) sensitivity and (4) recovery. To minimize the number of rules only two triangular membership functions are assigned per input variable and four triangular membership functions are used for the output variable. Fuzzy logic controller (FLC$_3$) observes four input variables and produce one output variable.

5.5. Fuzzy Modeling of Risk

Figure 8 illustrates the fuzzy associate memory map linking *threat level*, *vulnerability* and *asset value* for overall risk assessment. Four membership functions are used to represent each of the three input variables. Triangular membership values are used as membership functions. A fuzzy *if-then* rule may be formulated as follows:

IF *threat level* is **SIGNIFICANT** and *vulnerability* is **VERY HIGH** and *asset value* is **HIGH** ***THEN*** *Risk* is **HIGH**.

All the 9 input variable values and the output variable (risk assessment) are scaled between 0-1. The fuzzy *if-then* rules were formulated based on expert knowledge of the network model and associated risks.

6. Conclusions

This paper proposes a distributed intrusion prevention system, which is activated based on the predicted threat level and risk assessment of the protected assets. We focus on the distributed monitoring of intrusion attempts, one step ahead prediction of such attempts, and online risk assessment using fuzzy inference systems. Preliminary experimental results indicate that the proposed framework is efficient for real time distributed intrusion monitoring and prevention.

Future work may include parameter estimation based on real data, better HMM models, more states, continuous models, Kalman filtering and integration with mobile agents in a real network. We also intend to use supervised learning schemes to optimise the quantity and quality of the fuzzy *if-then* rules to improve the online computational performance etc.

References

[1] A. Abraham, C. Grosan, and C. Martin-Vide. Evolutionary design of intrusion detection programs. *International Journal of Network Security*, 4(3):328–339, 2007.

[2] A. Abraham, R. Jain, J. Thomas, and S. Han. D-scids: Distributed soft computing intrusion detection systems. *Journal of Network and Computer Applications, Elsevier Science*, 30(1):81–98, 2007.

[3] A. Årnes, K. Sallhammar, K. Haslum, T. Brekne, M. E. G. Moe, and S. J. Knapskog. Real-time risk assessment with network sensors and intrusion detection systems. In *International Conference on Computational Intelligence and Security (CIS)*, volume LNAI 3802, pages 388–397, Dec 2005.

[4] F. Gao, J. Sun, and Z. Wei. The prediction role of hidden markov model in intrusion detection. In *IEEE CCECE 2003: Canadian Conference on Electrical and Computer Engineering*, volume 2, pages 893–896, 2003.

[5] J. Jones. An introduction to factor analysis of information risk (fair). *Norwich Journal of Information Assurance*, 2(1):67, 2006.

[6] R. Khanna and H. Liu. System approach to intrusion detection using hidden markov model. In *Proceeding of the 2006 international conference on Communications and mobile computing*, pages 349–354, USA, 2006. ACM Press.

[7] E. Mamdani and S. Assilian. An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, 7(1):1–13, 1975.

[8] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Readings in speech recognition*, pages 267–296, 1990.

[9] L. Zadeh. Fuzzy sets. *Info. & Ctl.*, 8:338–353, 1965.