A-ToPSS: A Publish/Subscribe System Supporting Imperfect Information Processing

Haifeng Liu Hans-Arno Jacobsen

Department of Computer Science & Department of Electrical and Computer Engineering
University of Toronto
hfliu@cs.toronto.edu, jacobsen@eecg.toronto.edu

1 Introduction

A new data processing paradigm – publish/subscribe – is becoming increasingly popular for information dissemination applications. Publish/subscribe systems anonymously interconnect information providers with information consumers in a distributed environment. Information providers publish information in the form of publications and information consumers subscribe their interests in the form of subscriptions. The publish/subscribe system performs the matching task and ensures the timely delivery of published events to all interested subscribers. Publish/subscribe has been well studied and many systems have been developed supporting this paradigm. Existing research prototypes, include, among others, Gryphon, LeSubscribe, and ToPSS; industrial strength systems include various implementations of JMS, the CORBA Notification Service, and TIB/RV. All these systems are based on a crisp data model, which means that neither subscribers nor publishers can express uncertain information in subscriptions and publications, respectively. In this crisp model, subscriptions are either evaluated to be true or to be false, for any given publication.

However, in many situations exact knowledge to crisply specify subscriptions or publications is not available. In these cases, the uncertainty about the state of the world has to be cast into a crisp data model that defines absolute limits. Moreover, for a user of a publish/subscribe system, it may be much simpler to describe the state of the world with vague or uncertain terms. That means in an approximate manner.

In a selective information dissemination context, for instance, users may want to submit subscriptions about an apartment whose constraint on rent is "cheap". On the other hand, information providers may not have exact information for all items published. In a second-hand market, a seller may not know the exact age of a vase so that she can just describe it as an "old" vase, but can not describe it with an exact age. Temperature and humidity information collected by sensors are often not fully precise, but only correct within a certain error interval around the value measured. It would be more precise to publish such imprecise information, rather than a wrong exact value, if such publish/subscribe capabilities were possible.

For these reasons, it is of great advantage to provide a publish/subscribe data model and an approximate matching scheme that allows the expression and processing of uncertainties for both subscriptions and publications. There are five interesting cases according to the different combinations of subscriptions and publications with uncertainties. These are: 1. crisp subscriptions and crisp publications (conventional publish/subscribe), 2. approximate subscriptions and crisp publications, 3. crisp subscriptions and approximate publications, and 4. approximate subscriptions and approximate publications. A fifth case combines crisp and approximate constraints in subscriptions and publications. Models 2 to 5 constitute completely novel publish/subscribe system models not previously investigated. All existing publish/subscribe systems are based on a crisp data model that cannot process uncertainty in either publications or subscriptions. The only exception is A-ToPSS, the Approximate Matching based Toronto Publish/Subscribe System [3, 4], which has introduced a model that can express imperfect information, such as "cheap", "large", and "close to" in subscriptions and publications. In [3], only Case 2 was demonstrated and in [4] we have developed the theory to support all five cases. In this work we aim at demonstrating all the above cases and show approximate matching on real data sets based on an online auction scenario.

**Proceedings of the 30th VLDB Conference,
Toronto, Canada, 2004**

2 Publish/Subscribe Model

In a publish/subscribe system, two types of *imperfect information*, as classified by [5], may arise: *imprecision* and *uncertainty*. Imprecision relates to the content of a statement, which is used to refer to the incomplete information upon which publications or subscriptions may be based. The second type of imperfect information relates to the matching between publications and subscriptions, which we refer to as uncertainty. Uncertainty concerns the state of knowledge about the relationship between the world and the statement about the world. Often, *fuzzy sets* [2] are used to model imprecision and *possibility measures* [1] are used to model uncertainty. Our objective is to model imperfect information in subscriptions and publications and to define an approximate matching semantic for different cases of matching crisp with approximate subscriptions and publications. A detailed discussion of the theoretical framework of our approach can be found in [4]. Below, we summarize the key aspects of this framework to keep the demonstration description self-contained.

2.1 Publication and Subscription Model

Publications describe real world artifacts or states of interest through a set of attribute value pairs. When an exact value for a certain attribute is not available, we use a possibility distribution [1] to express the confidence that the attribute has a given value. As described in detail in [4], a publication is defined as a list of attribute function pairs as follows:

$$p = \{(a_1, \pi_1), (a_2, \pi_2), \cdots, (a_n, \pi_n)\}.$$

For example, a condo advertised for sale, may be described as "(size is $160m^2$) and (price is cheap)". The first attribute is *crisp*; it defines a definite value for size. The second attribute is *approximate*. It is qualified as cheap, which is defined as a function that designates the possibility of each value in the domain of discourse (i.e., all admissible rent values) as being "cheap". More formally, this publication can be represented by a set of attribute function pairs as follows.

$$P = \{(size, \pi_{60}), (price, \pi_{cheap})\}$$

The possibility distribution functions are depicted graphically in Figure 1. Subscriptions define users' interests and consist of individual predicates linked by (Boolean) operators. To continue our example from above, let us define a subscription for a family who is looking for an apartment with constraints on price, size, and condition. The subscription in natural language that specifies these constraints is as follows:

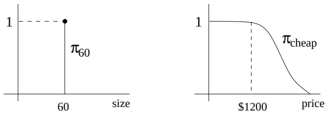


Figure 1: Possibility distributions for publication

S:	size	is	medium	AND
	price	is	*no more than* $1500	AND
	condition	is	*not* old.	

We use the following membership functions, depicted in Figure 2, to represent the concept of "≤ $1500","medium" and "old", respectively. So, the formal subscription language can be represented by

$$S = (size, \mu_{medium}) \wedge (price, \mu_{\leq \$1500}) \wedge (condition, 1 - \mu_{old})$$

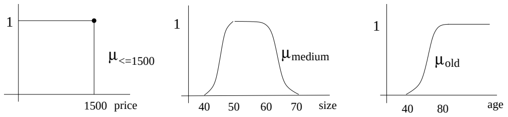


Figure 2: Membership functions for predicates

2.2 Approximate Matching

The semantics of matching subscriptions with publications is to measure the *possibility* and the *necessity* with which the publication satisfies the expectation expressed by a subscription. Based on possibility theory, we use a pair (Π_i, N_i) to denote the evaluation of the possibility and necessity of how a publication satisfies each predicate i (i.e., the match between μ_i and π_i) in a subscription. This measure is done by computing the intersection between μ_i and π_i as follows:

$$\Pi_i = \sup_{x \in D} \min(\mu_i(x), \pi_i(x))$$

$$N_i = \inf_{x \in D} \max(\mu_i(x), 1 - \pi_i(x))$$

Note that sup and inf have been chosen to represent the general case that μ_i and π_i are defined on an infinite domain. For the finite case, sup and inf are equivalent to max and min. The graphical interpretation of these measures is illustrated through a list of figures in Figure 3. With this matching semantic, users may be overwhelmed with a large number of slightly matching subscriptions, i.e., with a low degree of match. Thus, the approximate matching model introduces two

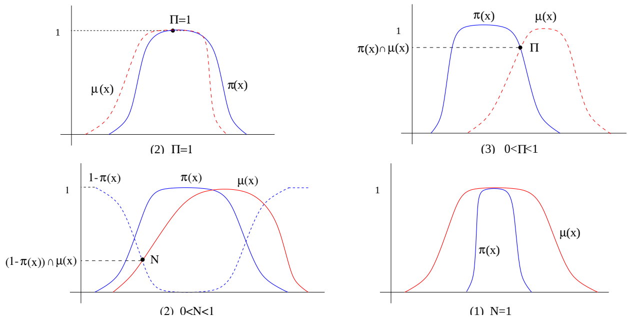


Figure 3: Possibility and necessity measures

parameters θ_Π and θ_N to control the tolerance of a match on a per-predicate basis for each subscription. These thresholds capture a users' satisfaction with the possibility and the necessity of how their interests are matched. Users' constraints are matched if both the possibility and the necessity degrees are larger than the thresholds θ_Π and θ_N. To illustrate these parameters, we adapt the apartment subscription example from above to:

$$\begin{aligned} \text{S:}\quad & (size, \mu_{medium}, 0.8, 0) && \wedge \\ & (price, \mu_{\leq 1500}, 1, 0) && \wedge \\ & (condition, 1 - \mu_{old}, 1, 0.3). && \end{aligned}$$

This subscription matches a publication as long as its `size` predicate matches a *medium* value with a degree of more than 0.8, which is the possibility threshold. The necessity threshold is irrelevant in this example (since it is 0). The `price` should be less than $1500, and the age predicate matches a `not-old` value with a possibility threshold of 1 and a necessity threshold of 0.3.

The general representation of a subscription is as follows:

$$S = R((a_1, \mu_1, \theta_{\Pi_1}, \theta_{N_1}), \cdots, (a_n, \mu_n, \theta_{\Pi_n}, \theta_{N_n}))$$

Here, R is the relation that aggregates the truth values of the individual predicates to an overall truth value for the subscription, S (e.g., a conjunction, a disjunction, or a normal form). The approximate matching problem can now be stated as follows. Given a set of subscriptions S and a publication p, the matching problem comes down to identifying all $s \in S$ such that s and p match with degrees greater than the thresholds defined on s by any subscriber.

3 Membership Function Mining

The A-ToPSS model offers its users great flexibility and leaves room for tuning a wide range of default parameters. It is often a challenge to select the right membership function parameterization, the exact number of membership functions to represent one dimension, the appropriate aggregation function or the right thresholds. However, A-ToPSS is used in a context where many subscribers (potentially millions) seek the right information. Consequently, much information about what defines certain concepts in specific domains is readily available, such as an "average understanding" of what constitutes a "cheap" price of a popular electronics gadget available in an online auction. If this information could be exploited, better default parameter choices could be determined for subscribers and publishers of such a system.

In A-ToPSS we experiment with a clustering-based approach that determines default value settings from past data (i.e., from past subscriptions and publications). We demonstrate our approach on real data traces that we have collected from an online auction site. The parameter estimation task is performed by an additional analyzer component on top of the core publish/subscribe system (see Figure 4 for details).

In the clustering analysis we do not differentiate between subscriptions or publications, but mine for default parameterizations of the membership functions underlying both entities. This is possible due to the link of a fuzzy set and a possibility distribution, explained in further detail in [4]. The mined parameterization is then used to provide default values for imperfect concepts. Currently, we focus on estimating the number of concepts that define one dimension and their function representation. For example, the dimension price, could be represented by the concepts "cheap", "fair" and "expensive" or by four concepts, adding, for instance, "luxurious". Each of these concepts is represented by a parameterized membership function, which can be adapted to a specific understanding by modifying its parameters.

4 Demonstration

The main challenge in applying publish/subscribe systems to selective information dissemination lies in the design of efficient algorithms that exhibit good scalability. At Internet-scale, such systems have to be able to process millions of subscriptions and react to thousands of events. The introduction of approximate matching increases the number of possible matches for a subscription, since more events will be matched, but to different degrees of match. To understand how the notion of approximate matching can help to better satisfy users' queries for information, we have taken an experimental approach that demonstrates how the results returned from crisp and approximate matching compare side-by-side. We have built a matching engine that can manage more than ten million subscriptions and process many events per second. We have integrated this with a web-server and an application server. Figure 4 represents the overall system architecture. All components are fully implemented. We will use an online auction application example to demonstrate our approximate matching scheme. Our

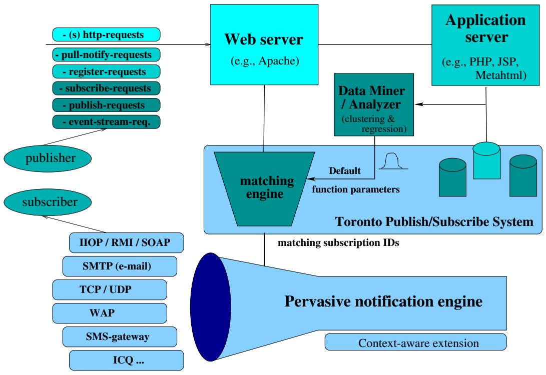
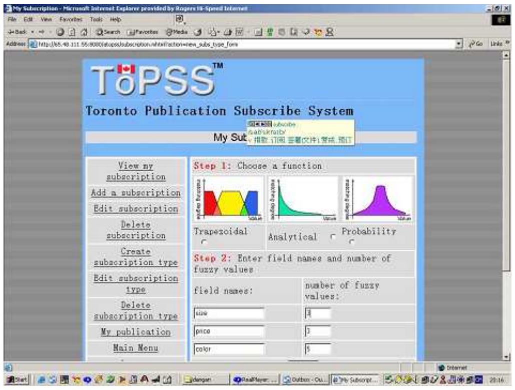


Figure 4: Overall system architecture

software demonstration will look at two aspects. One, an approximate publish/subscribe-based online auction driven by real world data (i.e., traces collected over the web from an auction site, referred to as Scenario 1) and two, an experimental comparison between the traditional publish/subscribe model and our model supporting imperfect information processing and approximate matching (referred to as Scenario 2). The former serves to demonstrate the viability of our model in a real world context and the latter serves to show the difference between crisp and approximate publish/subscribe.



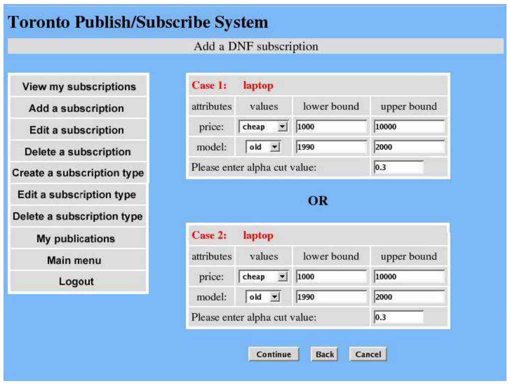
Step 1: Choose a function

○ Trapezoidal ○ Analytical ○ Probability

Step 2: Enter field names and number of fuzzy values

field names:	number of fuzzy values:
size	3
price	3
color	5

Figure 5: The power user's interface for defining membership functions



Add a DNF subscription

Case 1: laptop

attributes	values	lower bound	upper bound
price:	cheap	1000	10000
model:	old	1990	2000
Please enter alpha cut value:			0.3

OR

Case 2: laptop

attributes	values	lower bound	upper bound
price:	cheap	1000	10000
model:	old	1990	2000
Please enter alpha cut value:			0.3

Figure 6: The interface to define a DNF subscription

Scenario 1 will demonstrate the flexibility in expressing publications and subscriptions in the approximate model. A subscriber chooses among a family of functions to represent imperfect information. The system will provide default parameters for selected functions based on mining of past data seen (subscriptions and publications). We model past data from the collected trace data. Figure 5 shows a screen shot of the subscription entry panel of our system, where a user can view and adapt the default membership functions representing her subscriptions. The default parameters are set by the analyzer component and will adapt over time. Expressiveness is further demonstrated by allowing subscribers to submit subscriptions in disjunctive or conjunctive normal forms (DNF/CNF), a unique feature of our matching engine. To date, most existing publish/subscribe research focuses on conjunctive subscriptions only. Figure 6 illustrates the panel for a DNF subscription. To demonstrate Scenario 2, a panel to dynamically control publication representations is available, which concentrates on integer interval values, in which case the publication is of the form $(attribute_i, [a, b])$. By changing the range of the interval $[a, b]$, the number of matched subscriptions can be influenced. This effect can be observed in a monitoring panel and compared to crisp matching. Figure 7 illustrates the numbers of matched subscriptions, comparing crisp with approximate matching. The intuition is to provide default settings for values, a and b that accommodate pessimistic, optimistic, or indifferent subscribers (i.e., those who have a low tolerance for matches versus those who fear to miss a good match.) Scenario 2 will allow us to also demonstrate the effects of the representation of membership functions, the influence of different aggregation functions (i.e., min, max, or product in subscription evaluation), and the effects of various thresholds. It is difficult to absolutely quantify subscriber satisfaction, we therefore expect the software demonstration to further illustrate pros and cons of our approach.

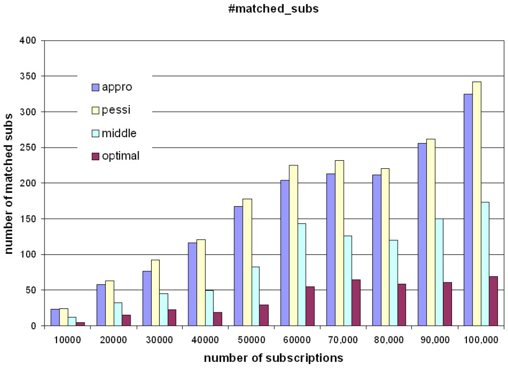


Figure 7: Number of matches for different cases

References

[1] D. Dubois and H. Prade. *Possibility Theory: An Approach to Computerized Processing of Uncertainty.* Plenum Press, New York, 1988.

[2] G. J. Klir and T. A. Folger. *Fuzzy Sets, Uncertainty, and Information.* Prentice Hall International Editions, 1992.

[3] H. Liu and H.-A. Jacobsen. A-TOPSS – a publish/subscribe system supporting approximate matching. In *28 th International Conference on VLDB*, Hong Kong, China, 2002.

[4] H. Liu and H.-A. Jacobsen. Modeling uncertainties in publish/subscribe system. In *20th International Conference on Data Engineering*, Boston, USA, 2004.

[5] P. Smets. *Imperfect information: Imprecision-Uncertainty, Uncertainty Management in Informaiton Systems: From needs to Solutions.* Kluwer Academic Publisher, 1977.