Proceedings of the 5th WSEAS Int. Conf. on SIMULATION, MODELING AND OPTIMIZATION, Corfu, Greece, August 17-19, 2005 (pp633-638)

Robust methods for Databases and robotics

CARLOS RODRÍGUEZ LUCATERO, MICHEL de ROUGEMONT & RAFAEL LOZANO ESPINOSA

Departamento de Computación

ITESM Campus Ciudad de México
Tlalpan Ciudad de México, México, C.P. 14380
MEXICO & LRI Université Paris XI

Abstract: - The goal of this article is to propose the development of robust methods for the processing of different data sources that appear in the integration of Databases and in the data fusion in robotics.In the Databases field, one wish to classify uncertain data and to answer queries in an approximated way in the case that data sources are incoherent. Our interest is based on the fundamental problems in the domain of XML data as well as in the data flow.Concerning mobile robotics we would like to compare robot strategies in uncertain environments and to learn good strategies in that situations. From one side we have interest on exploring some property proof and learning aspects, and by the other side, we are interested on game theory equilibrium techniques.

Key-Words: - Database, XML, Robot Motion Planning, Complexity, Markovian strategies.

1 Introduction

The robustness problem is fundamental in the information and communication domain given that it introduce a physical element, the lack of precision and the uncertain aspects that impose new scientific problems. The classic computational techniques are often not very robust to data errors and the good notions of approximation are frequently hard to pose. The relation between approximation and computational complexity is a recent area that have allowed to clarify some computer science issues since ten years ago.The error correcting codes as well as information theory fields have been interested since Shannon in studying the lack of precision and the algorithmic techniques that allow to transmit information in a noisy channel. It is only till our days that the fundamental relations between computational complexity and correcting error codes have emerged. The Spielmans' work as well as those related with turbo codes have shown the interest of that by means of randomized algorithms. The links with the cryptography field have allowed to better understand certain robustness notions. The property testing notion [4] [5]). introduced in [1,2] is one of the bases for our research. This notion is related with computational learning. The proposal is to study fundamental problems and to show how to apply to the robotics field some results obtained in the domain of databases on the Web.

1.1 Fundamental problem

The property testing notion introduced in the context of program testing is one of the foundations of our research. If **K** is a class of finite structures and *P* is a property over **K**, we wish to find a **Tester**, in other words, given a structure *U* of **K**:

- It can be that *U* satisfy *P*.
- It can be that *U* is ε-far from *P*, that means, that the minimal distance between *U* and *U'* that satisfy *P* is greater than ε.
- The randomized algorithm runs in O(ε) time independently of *n*, where *n* represent, the size of the structure *U*.

For instance, if **K** is the class of graphs and *P* is a property of being colorable, it is wanted to decide if a graph *U* of size *n* is 3-colorable or ε-far of being 3-colorable, i.e. the Hamming distance between *U* and *U'* is greater than $\varepsilon . n^2$. If **K** is the class of binary strings and *P* is a regular property (defined by an automata), it is wished to decide if a word *U* of size *n* is accepted by the automata or it is ε-far from being accepted, i.e., the *Edition distance* between *U* and *U'* is greater than ε.n. In both cases, it exists a **tester**, that is, an algorithm in that take constant time, that depends only on ε and that decide the proximity of these properties. In the same way it can be obtained a **corrector** that in the case that *U* does not satisfy *P* and that *U* is not ε-far, finds a structure *U'* that satisfy *P*. The existence of **testers** allow us to approximate efficiently a big number of combinatorial problems for some privileged distances. As an example, we can estimate the distance of two words of size *n* by means of the *Edition distance* with shift, we mean, when it is authorized the shift of a sub-word of arbitrary size in one step. To obtain the distance it is enough to randomly sample the sub-words between two words, to observe the statistics of the random sub-words and to compare with the L_1 norm. In a general setting, it is possible to define distances between automata and to quickly test if two automata are near knowing that the exact problem is NEXPTIME hard.

2 Application to the Database field

We are interested in the class of ordered and unordered labeled trees and we look for classification as well as to query them.

2.1 Testers and correctors in XML

The XML data with their DTDs and structured data built by different agents can use the same DTDs or two near DTDs. As a consequence we will have to define three notions:

- A document *T* is near to a document *T'*.
- A document *T* is near to a DTD.
- Two DTDs are near.

We propose to use the *Edition distance* of trees with a shift to define the two first notions, as have been defined before and the homomorphism approximation for the third notion. In the same way, it is possible to propose a way to generalize the **testers** and the **correctors** in the case of languages of regular trees defined by tree automata. The existence of a **tester** for regular trees has been shown in [7] for the *Edition distance* with shift and it has been shown that it is possible to decide in constant time if *T* is ε-far or not from $M_1...M_i...M_k$. The principle over it is based such a **tester** is to sample randomly the sub-trees DOM (Document Object Model) of a document: if it is ε-far from a DTD M_i, then one of the sample satisfy a local property that is easy to verify with a big certainty. A corrector is presented in [6] for the Edition distance without shift that is available for downloading at http://www.lri.fr/~mdr/xml and allows to find a T' that is valid for M and always near to T and when T is k-approximated to M, all that in linear time. We propose to generalize the corrector to the Edition distance with shift in order to fix errors on the moment that the distance is at a distance ε.n and no more than a constant k. The local corrections in each node use an essential function: given a word W and a regular expression R, find the nearest word W' (R) to the Edition distance with shift over the words. This problem is solved with the help of the corrector for the regular words. It is possible, with these tools, to classify the Web by a set of DTDs $M_1...M_i...M_k$ using over each document the tester in constant time. Afterwards, and if the test is positive, the corrector will be used in linear time over the same document. The main problem is the big variety of DTDs for the same type of document, in particular due to linguistic variations.

The following DTDs define documents of book type and are near in some sense we wish to define.

DTD book:
<?xml version=”1.0”><!ELEMENT book (chapter*,title,author) > <!ELEMENT chapter (title,para*)>
<!ELEMENT title (#PCDATA)> <!ELEMENT para (#PCDATA)> <!ELEMENT author (#PCDATA)>

DTD libro:
<?xml versión=”1.0”><!ELEMENT libro (capitulo*,editor,autor,tiulo) >
<!ELEMENT capitulo

(titulo,parra*)><!ELEMENT titulo (#PCDATA)>
<!ELEMENT parra (#PCDATA)><!ELEMENT autor (#PCDATA)> <!ELEMENT editor (#PCDATA)>

2.2 Distance between DTDs

We define the distance between two regular expressions for the case in that it is used or not the same language. It is defined the distance between two regular expressions r and t over the same language L as:

$$Dist(n,r,t) = Min_{w\, in\, r, w'\, in\, t, |w|=|w'|=n}\ dist(w,w')$$

The distance between two DTD's M_1 and M_2 over the same language L is given by the function:

$$Dist(n,M_1,M_2) = Min_{F\, in\, M\_1, F'\, in\, M\_2, |F|=|F'|=n}\ dist(F,F')$$

Two DTDs over the same language are at a distance O(1), or constant distance if Dist(n,r,t)= O(1) and we are interested in those DTDs that are near. If the regular expressions *r* and *t*, or the DTDs, use different languages L_1 and L_2, we generalize this definition considering all the partial functions π between L_1 and L_2 taking the Minimum for all π at distance between π(*r)* and *t* .

Two regular expressions *r* and *t*, are isomorphic if it exist π such that π(*r)=t* . Two regular expressions *r* and *t*, in Unary Normal Form, it means using * over only one character, such that |*r*| < |*t*| are homomorphic if it exists a partial application π between L_1 and L_2 such that the distance between π(*r)* and *t* be O(1).

Then we can generalize this definition between two reduced DTDs that use different languages. For a bag <*a*>, let it be DNF(*a*) the regular expression that define *a*. Lets suppose that | L_1 | < | L_2 | and lets call *a*, *recursive,* if *a** belongs to DNF(*a*) or if *a* is in a loop of the dependency graph whose bags are the nodes and the edges link a bag with all those that appear in his DNF. Two DTDs M_1 and M_2 with roots r_1 and r_2 are homomorphic if it exists a partial application π between L_1 and L_2 such that :

(1) if *a* is recursive in M_1 then π(*r)* is recursive in M_2 and π(DNF(*a*)) is isomorphic to DNF(π(*a*))

(2) if *b* is not recursive in M_1 then π(DNF(*b*)) is homomorphic to DNF(π (*b*)),

(3) π(r_1)= π(r_2).

Then we can show that two regular expressions *r* and *t* homomorphic are at distance O(1) and that two homomorphic DTDs are at distance O(1).

Then we can imagine a corrector that enumerate all the applications π, but it will be inefficient, however always in linear time. An exponential of a constant as a function on the size of the alphabet appears. We can generalize as well the corrector, guessing in a bottom-up fashion an application π, hopping to eliminate this exponential factor.

Generalized Corrector. Input *: F of DTD M_1 and one DTD M_2*. Output *: F' valid for M_2*

(1) To guess an partial application π, in the bottom-up analysis an to estimate in each step the Edition distance of the trees.

(2) If such distance is greater than k, consider another partial application π.

(3) *If we reach to root r with π and a distance less than k, then fix* π(F).

The theoretical problem stated is the existence of a tester and corrector for the approximated equality of two regular expressions. The exact solution to the problem is NEXP complete. We wish to study the theoretical problem and the implementation of a generalized corrector that allow us to estimate distances of documents for near DTDs and its generalization to RELAX NG and W3C scheme.

3 Document annotation and query languages

The annotations are sets of terms that we associate in a compositional fashion. It can be taken as a starting point the compositional approach of the annotations developed in [5], being the idea that the environment near to a given node serves as base to a inference mechanism of the annotation of such node.

We will try to associate not only one composed annotation but a set of annotations endowed with a distance between annotations. A possible starting base is to take into account the metrics proposed for measuring the distance between two terms in an organized taxonomy according to a subsumption relation. As an example the similarity of Wu and Palmer express the distance between two terms A and B as :
$d(A,B) = 2*Depth(C)/(Depth(A)+Depth(B))$

where C is the nearest common predecessor to A and B. We will try to extend such kind of metrics by one hand to the set of terms (annotations), and by the other to a hierarchical structure where each node should be annotated.

It will be studied the possibility of applying such kind of compositional metrics to documents annotated by different terminologies for which we dispose however of "give articulations" establishing relations among terms. This extension is particularly interesting for distributed environments of document management , where the information system contents is shared among many organizations using different classification modes (for instance an eLearning environment).

Let T be a taxonomy and D a document. In [5], the notion of annotated document for T is defined and a RDF annotation is associated to each document. We can in these way associate to a query (a list of words) a set of sub-documents, i.e. URLs and Xpath paths.

If we consider a taxonomy set T_1,...T_n that use many natural languages, it is wished to generalize the precedent scheme introducing the approximated annotation notion for T_1,...T_n. If an annotated document is valid for T in his creation, it is no more without doubt for T_1,...T_n but admits a distance for each one of the T_i. In the case of an structured searching engine, this one sets T_1,...T_n and accumulates the documents already annotated for the partially known taxonomies.

We wish to define the distance of an annotated document D of a taxonomy T and it can be chosen the Edition distance (and his variants) among the annotation graph and each one of the T_i. Then it will be possible to associate to a query the set of nearest sub-documents ordered by increasing distance.

3.1 Integration of data sources

In the Data Bases framework on the Web, we can consider independent sources that for a given query produce different results. We can consider sources S_1 , S_2 ,…, S_p that we must fusion for giving a robust answer under a given criteria. One possible approach is to associate qualities to each source, a numerical vector, that precise for each source parameters as : the precision of the data, his most recent update, his degree of specialization ,….

Then a user can precise his preferences as another vector of the same kind. Then we can determine how to do the sources fusion, it means, to associate numerical values λ_1 , λ_2 ,…, λ_p such that $\Sigma_i \lambda_i =1$ and the answer $Q= \Sigma_i Q_i$ where Q_i is the answer to S_i , that maximize a global quality criteria.

Under a more elaborated approach, each source produce only one data view. Then we can consider a game with many players where each source be a player which utility be the λ_i of just mentioned procedure. What is the equilibrium of this game ? How can be compared a fusion solution with an equilibrium or with an approximated equilibrium ? Then we can consider the data integration as the equilibrium of a physical process, that must be robust and easily calculable.
In the processes approach, it starts from a solution as the just mentioned fusion and it is posed the question about what is the game which equilibrium is such a solution. Such mechanisms have been proposed in [4] by J. Kleinberg, C. Papadimitriou and P. Raghavan and pose big number of algorithmic problems.

4 Applications to Robotics

The mobile robotics framework is more complex because we should process data flows provided by the captors under a dynamic situation, the robot moving, taking into account two kind of uncertainty :

- The captors lack of precision and subject to errors

- The robot is subject to deviations of his trajectory as any mechanical object.

The data flow provided by the captors state a similar problematic to that of the databases. The robot should make the information sources fusion to determine his strategy of action and movements. Some sources, called bags, allow to the robot to self locate geometrically or in his state graph. While the robot execute his strategy, it is subject to movement uncertainties and then should find robust strategies for such a uncertainty source. The goal is achieve the robustness using the data flow of the captors integrated to the strategies. We consider the classical form of simple Markovian strategies. In the simplest version, a Markov chain, MDP, is a graph where all the states are distinguishable and the edges are labeled by actions L_1 , L_2 , L_3 ,…, L_p . If the states are known only by his coloration in k colors C_1 , C_2 , C_3 ,…, C_k, two estates having the same coloration are undistinguishable and we are talking about POMDP (Partially Observed Markov Decision Process). A simple strategy σ is a function that associate an action simplex to a color among the possible actions. It is a probabilistic algorithm that allows to travel the state graph with some probabilities. With the help of the strategies we look for reaching a given node of the graph from the starting node (the initial state) or to satisfy temporal properties, expressed en general in LTL . For instance, the property *C_1 Until C_2* that express : we can reach a node with label C_2 preceded only by the node C_1 . Given a property θ and a strategy σ, let Prob $_\sigma$(θ) be the probability that θ is true over the probability space associated to σ.

4.1 To compare two strategies over MDPs and POMDPs

Given a POMDP M, two strategies σ and π , it can be looked for estimating the Prob $_\sigma$(θ) and to know if Prob $_\sigma$(θ) > Prob $_\pi$ (θ). If Prob $_\sigma$ (θ) >b, it is frequent to test such a property while b is not very little with the aid of the path sampling according to the distribution of the POMDP. In the case that b-ε < Prob $_\sigma$(θ) < b , it can be looked for a corrector for σ, it means, a procedure that lightly modify σ in such a way that Prob $_\sigma$ (θ) >b. It can be modified too the graph associated and in that case, we look for comparing two POMDPs.

4.2 To compare two MDPs and POMDPs

Let be M_1 y M_2 two POMDPs, we want to compare this POMDPs provided with strategies σ and π in the same way as are compared two automata in the sense that they are approximately equivalent (refer to section concerning distance between DTDs). How can we decide if they are approximately equivalent for a property class? Such a procedure is the base of the strategy learning. It starts with a low performance strategy that is modified in each step for improvement. The Tester, Corrector and learning algorithms notions find a natural application in this context.

4.3 To compare strategies with uncertainty in the mouvements

One of the specificities of mobile robotics is to conceive robust strategies for the movements of a robot. As every mechanical object, the robot deviates of any previewed trajectory and then it must recalculate his location. In some preceding articles [12,13,14,8], we have introduced the uncertainty model of the deviations :

At the exeecution of an action L_i comanded by the robot, the realization will follow L_i with probability p, an action L_{i-1} with probability (1-p)/2and an action L_{i+1} with probability (1-p)/2.

This new probabilistic space induce robustness qualities for each strategy, in other words, the Prob $_\sigma$ (θ) depends on the structure of the POMDP and on the error model. Then it can be formulated the same questions that have been posed before : how to evaluate the quality of the strategies, how to test properties of strategies, how to fix the strategies such that we can learn robust strategies. We can consider as in [9,10,11] that the robot plays a game against nature that is similar to a Bayesian game. The criteria of robust strategy are similar to those of the direct approach.

5 To compare strategies of many robots and conclusions

In this situation, the robots can be or not cooperative. If they do not cooperate, they are classical players that try to do different tasks. Then it becomes interesting for a robot to find a robust strategy that lends him to an equilibrium. If the robots cooperate, they try to execute a complex task. The cooperative game theory define strategies for this end, as the Shapleys' value. We want to know the robustness of such strategies and to pose the classical questions in this context.

References:

[1] R. Rubinfeld and M. Sudan. Robust characterizations of polynomials with applications to program testing, SIAM Journal on Computing, 1996.

.

[2] O. Goldreich and S.Goldwasser and D.Ron.Property Testing and Its Connection to Learning and Approximation.IEEE Symposium on Foundations of Computer Science, 1996.

[3] N. Alon and M. Krivelevich and I. Newman and M. Szegedy. Regular languages are testable with a constant number of queries, IEEE FOCS, 1999.

[4] J. Kleinberg and C. Papadimitriou and P. Raghavan.
On the value of private information, Tark conference, 2001.

[5] B. Gueye, P. Rigaux, N. Spyratos, Annotation automatique de documents XML, Actes des journées 'Extraction et gestion des connaissances' (EGC'04), Clermont-Ferrand, France,January,2004.

[6] M. de Rougemont. The correction of XML data, ISIP'03 (First Franco-Japanese Workshop on Information Search, Integration and Personalization), Sapporo 2003.

[7] Magniez F. , de Rougemont, M. Property testing of regular tree languages, ICALP 2004.

[8] M. de Rougemont, C. Schlieder, Spatial navigation with uncertain deviations, AAAI Conference on Artificial Intelligence, 1997.

[9] S.M. La Valle, David Lin, Leonidas J. Guibas, J.C. Latombe & Rajeev Motwani,Finding an Unpredictable Target in a Workspace with Obstacles,IEEE International Conference on Robotics and Automation, 1997.

[10] R. Murrieta-Cid, H.H. González-Baños , B. Tovar, A Reactive Motion Planner to Maintain Visibility of Unpredictable Targets,IEEE International Conference on Robotics and Automation, 2002..

[11] C. Rodríguez Lucatero, A. de Albornoz Bueno, R. L. Espinoza, A game theory approach to the robot tracking problem, ISPRA04 Salzburg Austria.

[12] M. de Rougemont, J.F. Diaz-Frias. A theory of robust planning. IEEE Int. Conf. Robotics and Automation, 1992

[13] M. de Rougemont, J.L. Marion and C. Rodriguez. The evaluation of strategies in motion planning with uncertainty, IEEE Int. Conf. Robotics and Automation, 1994

[14] D. Burago , M. de Rougemont, A. Slissenko. The complexity of motion planning under uncertain deviations, Theoretical Computer Science 1996.