CONTENTS

Invited Lectures

Knowledge in Computational Linguistics

Knowledge Processing and Discovery

Knowledge in Software Engineering

Knowledge in Distributed Computing

KNOWLEDGE

PROCESSING AND DISCOVERY

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 95–98

THE AUTONOMOUS ROBOTIC TANK (ART): AN INNOVATIVE LEGO MINDSTORM NXT BATTLE VEHICLE

RADU ŢURCAŞ, OANA IOVA AND ZSUZSANNA MARIAN

Abstract. The Autonomous Robotic Tank (ART) is a miniature tank built from Lego Mindstorm NXT parts that scans its surrounding and detects targets of an *apriori* set colour with the help of an ultrasonic sensor, a webcam and a flashlight. ART shoots projectiles towards the identified targets, with a slingshot-based firing system.We have developed two algorithms named *FirstTry* and *SecondTry* that can be used in different situations by the tank. The *SecondTry* algorithm is a more complex adaptation of the *FirstTry* algorithm. Statistics show that *SecondTry* algorithm is better than *FirstTry*.

1. Introduction

Robots are taking the place of humans in many industrial processes, for faster and more accurate production. In medical care, robots such as ROBODOC can perform orthopaedic surgeries [1]. Robots are also an important part of education. A commonly used educational robot is the Lego Mindstorm NXT, which can be used both by children and by specialists, depending on the complexity of the task. Also, robots are often designed to replace humans in life threatening situations. For example, they can be used to explore unknown terrain, search for wounded people in poorly accessible areas or can engage in military action.

We have built, designed and programmed a Lego Mindstorm NXT robot that acts as a miniature version of an autonomous tank, using tracks for mobility and firing projectiles towards targets. The robot searches for red-coloured objects according to an algorithm and shoots towards them with plastic balls as projectiles. Based on its major characteristics, the robot was named Autonomous Robotic Tank (ART). We implemented two different algorithms, termed *FirstTry* and *SecondTry*. Statistics show that the robot performs very well, with higher efficiency for the *SecondTry* algorithm.

The paper is organized as follows: in Section 2 we describe several robots that share features with ART. In Section 3 we offer a description of the design and engineering of ART. Finally, we give some concluding remarks in Section 4.

2000 *Mathematics Subject Classification.* 68T05,91E45.

2. Related work

The concept of an autonomous tank or war machine can be identified in several other robots, such as Ripsaw MS1, The Mercenary, Mad Max, or Stryker. The latter is an army tank that is able to detect obstacles and to operate based on the information gathered by its on-board sensors and by the lead vehicle of the convoy [2]. The Ripsaw MS1 is an Unmanned Ground Vehicle that can reach high speeds and advance over large obstacles. However, it is not autonomous [3]. The Mercenary autonomously tracks and shoots at players with paintballs [4]. Mad Max is an autonomous tank that performs bump sensing, obstacle avoiding, target finding and triggering [5].

3. Autonomous Robotic Tank

The goal of our project was to create a miniature Lego version of a tank that can autonomously select its targets from a multitude of environmental stimuli and fire projectiles only at them.

3.1. **Components.** The Lego Mindstorm NXT kit offers a wide variety of components that can be assembled in different ways, depending on the task to be performed. The central piece is the NXT brick that serves as the "brain" of the robot [6]. It is accompanied by different sensors, servo motors, and technical pieces. The programming language used was Bricxcc/NXC [7].

3.2. **Technical details.** ART has tracks, one on each side, for better stability and easier manoeuvring. A third track is part of the shooting system, which acts on the basic principle of a slingshot. This track is located inside the main body of the robot and has a pin attached to it. When the pin reaches the upper side of the track, it moves a Lego piece that functions as a striker. The striker has a rubber band attached. Once the pin reaches the end of the track, the striker slips and is pushed forward with force by the rubber band, hitting a projectile that flies towards the target. Extra ammunition is kept in a so-called charger space. Eight projectiles are stacked on top of each other in the frontal part of the robot and reloading is automatic. The ultrasonic sensor serves to identify surrounding objects so that the robot will be able to avoid clashing into them. The NXT V1.1 web cam is used to detect the targets of a specific colour. A flashlight serves to emit light that can cover only a small surface in the direction of the projectile, parallel to the ground. Only the lit part of the target object is recorded as the target. In this way, the detection of fallen targets that lie on the floor is avoided because the beam of light will not hit targets that are still in the webcam range.

3.3. **Algorithm.** In the following sections, we propose two algorithms, named *FirstTry* and *SecondTry*, for guiding the actions of ART in the field.

Image processing. The camera can track a line or a maximum number of 8 objects which may have one of 8 different, user defined colours. The choice for ART was the object tracking mode. The NXTCamView program allows capturing a picture, selecting a colour from it and uploading it to the camera. As a result the camera will only detect the uploaded colour, a hue of red recorded from the lit part of the

target. NXTCamView has another feature, called tracking, which simulates the way the camera works: it represents the objects the camera "sees" as if they were composed of rectangles with the X and Y coordinates from the upper-left corner, and records the height, width and colour of the object.

Robot actions. In the *FirstTry* algorithm, the first contact to the environment is established by measuring the distance to an object in front of the robot. As long as the distance is greater than the reliable range of the camera (40 cm) the robot keeps going forward. When there is something found in its range, ART verifies if the object is a target by reading the number of rectangles detected by the camera. If this number is greater than zero, than there is a target in front of the robot and shooting occurs. If the object is not a target, it makes a random 90 rotation to the left or to the right. The robot follows this algorithm until it runs out of balls. If the number of targets is smaller than the number of balls, the robot keeps searching for targets for surveillance. This algorithm is fairly simple, but it has an advantage: it makes no assumptions about the size of the territory or about the position of the robot inside it. This is important because it can apply to any territory.

An improved version. SecondTry is an improved version of the algorithm. When the robot is stopped by an obstacle, it starts turning around and in the meantime checks if there are objects around it. If ART senses something closer than 10 cm it goes back for 0.5 seconds in order to avoid contact and protect the integrity of the camera or the flashlight. When a target is found at less than 40 cm from the robot, it stops and shoots. If the camera still detects a target in front of the robot, a second projectile is shot. To avoid the entrance in a redundant cycle, the rotation is no longer initiated if the robot has already made a 360 rotation without stopping. In this case, a while loop makes the robot turn 150 if there is something in its way. Based on a series of empirical experiments, we observed that 150 is the optimal value to be used. This turn is important because at the end of a round in which no targets were detected and no stops were made, the robot might face an obstacle.

3.4. **Numerical Experiments.** In order to verify that *SecondTry* is indeed an improved version of the *FirstTry* algorithm we tested both algorithms in the same conditions. The test field was a square with dimension 180 cm x 180 cm. Four red and two blue targets were used for the tests, each 30 cm in height and 6.5 cm in diameter. For each algorithm, thirty tests were conducted. The results showed that for the *FirstTry* algorithm the average time needed to hit all four targets is 318.4 seconds (standard deviation 191 seconds), while for the *SecondTry* algorithm the time is just 134.4 seconds (standard deviation 54.6 seconds). The 95% confidence intervals for the results of the two algorithms do not overlap. In conclusion, the *SecondTry* algorithm is an improvement of the *FirstTry* algorithm.

3.5. **Discussion.** Taking into consideration the statistical analysis, ART can be considered a success, but many improvements could still be achieved. Technically, using more resistant parts and finding a more firm way of coupling the flashlight to the robot would improve its stability and the accuracy of the firing. A more specialized webcam would improve colour recognition of the targets and may allow tracking. In

the case of other examples such as Madmax or the Mercenary (Section 2), it is enough to rotate the gun and the camera to follow targets, while in our case ART should turn as a whole, since its firing system is an integral part of the robot's main body.

4. Conclusion

The Autonomous Robotic Tank resembles a tank, both in appearance and in function. However, it is not the epitome of a regular tank, as it has a large charger rising vertically and also a light-emitting device, a webcam and a Brick for target recognition. The latter components and the two implemented algorithms make ART autonomous. By functioning according to a predefined algorithm, the robot can work in the field independently of the presence of human beings. There is increasing emphasis on the principle of autonomy because the purpose of robotics is to protect and improve human lives.

Acknowledgements. This work was supported by the grant CNCSIS IDEI 543 coordinated by Prof. Dr. Mihai Oltean. We thank him for the technical support, guidance and advice offered during the entire project.

References

[1] M. Tavakoli, R. V. Patel, M. Moallem and A. Aziminedjad. Haptics for Teleoperated Surgical Robotic Systems. World Scientific, 2008.
[2] W. M. Solis. Realistic Deployment Timelines Needed for Army Stryker Brigades. GAO, 2003.
[3] Ripsaw MS1 http://www.defensetech.org/archives/004569.html, last visited on 02.03.2009.
[4] The Mercenary http://zero-op.com/mercenary, last visited on 02.03.2009.
[5] Madmax http://www.mil.ufl.edu/imdl/papers, last visited 05.03.2009.
[6] M. Gasperi, P. Hurbain, I. Hurbain, Extreme NXT: Extending the LEGO MINDSTORMS NXT to the Next Level, Apress, 2007.
[7] Bricxcc / NXC (code) http://bricxcc.sourceforge.net/nbc, last visited on 07.03.2009.

Babeş Bolyai University, Cluj-Napoca, Romania
E-mail address: {turcash_radu,oana_iova,marianzsu}@yahoo.com

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 99–102

DISTRIBUTED ASYNCHRONOUS COLLABORATIVE SEARCH

ANCA GOG, CAMELIA CHIRA, AND D. DUMITRESCU [1)]

Abstract. A collaborative search model (DACE) relying on the interaction and coevolution of spatially distributed individuals is studied. Population is organized in three agent societies (Local, Far and Global) engaged in asynchronous search process. Six variants of DACE characterized by the presence/absence of fitness ordering, local recombination and agent-based search are considered. The most efficient model is identified for a set of test problems.

1. Introduction

Connections between population geometry and individual interactions in evolutionary search are explored with the aim of developing a model able to efficiently address complex optimization/search problems. The *Distributed Asynchronous Collaborative Evolutionary (DACE)* model relies on asynchronous population model and search operators. The DACE population is endowed with a topological structure guiding selection and search processes [1,2]. The DACE model is a very general distributed search scheme admitting several instances able to cope with particular fitness landscapes (environments). Different strategies for recombination and population geometry are investigated and studied in several numerical experiments.

2. Distributed Asynchronous Collaborative Evolutionary Model

The *Distributed Asynchronous Collaborative Evolutionary (DACE)* model integrates evolutionary optimization with emergent behavior generated by agent-based interactions and recombination strategies facilitated by a certain population topology.

In order to ensure a flexible search process in solving very difficult problems, DACE individuals are identified with agents [3] having the objective of optimizing a fitness function. This objective is pursued by communicating with other individuals, selecting a mate for recombination based on individual strategies and performing mutation. Individuals can exchange information regarding specific environment characteristics (such as the current fitness value) and specific messages (such as request and inform) for establishing recombination strategies.

2000 *Mathematics Subject Classification.* 68T20, 68Q25.
Key words and phrases. evolutionary algorithms, asynchronous search, collaborative strategies, agent-based search.

The model implies three societies of agents (individuals). LIA (Local Interaction Agents) behaviour emphasizes local search while FIA (Far Interaction Agents) individuals are able to guide the search towards unexplored regions. The GIA (Global Interaction Agents) society focuses on the global exploration of the search space realizing the connection between the LIA and FIA societies. Population dynamics emerges through the recombination of individuals from different societies and a dominance principle - an offspring belongs to the class of the dominant parent. The dominance probability represents the (probabilistic) membership degree of an offspring to the society LIA (FIA) when one of the parents is a GIA individual.

The model uses an asynchronous search scheme. Individuals from a certain layer are updated through proactive recombination and are involved in forthcoming recombination and mutation processes within the same epoch. Details about this search model can be found in [1,2].

3. DACE Strategies

In order to select the most efficient DACE strategy, different DACE variants have been considered along the following three main ideas: order of individuals based on fitness, recombination strategies between individuals belonging to different societies and agent-based behaviour. Corresponding to these strategic alternatives, six different variants of the DACE model are studied as follows:

- *DACE_UL (DACE Unordered Local)* and *DACE_UG (DACE Unordered Global)*, where individuals are randomly distributed on layers and individuals from LIA and GIA accept as mates only individuals from the same society, respectively GIA individuals as well;
- *DACE_OL (DACE Ordered Local)* and *DACE_OG (DACE Ordered Global)*, where individuals are arranged according to their fitness and individuals from LIA and GIA accept as mates only individuals from the same society, respectively GIA individuals as well;
- *DACE_WL* and *DACE_WG (DACE Without societies and with Local, respectively Global recombination)*, where individuals are arranged according to their fitness but they are not structured in societies; mates are selected for recombination only from their local neighborhood, respectively on a global basis from the entire population.

4. Numerical experiments

A set of six benchmark functions [4] is engaged for these numerical experiments as follows: f_1 - Shifted Sphere Function, f_2 - Shifted Rosenbrock's Function, f_3 - Shifted Rotated Ackley's Function With Global Optimum On Bounds, f_4 - Shifted Rastrigin's Function, f_5 - Shifted Rotated Rastrigin's Function and f_6 - Shifted Rotated Weierstrass Function.

The algorithms have the following parameters: a population consists of 8*8 (64) individuals, the mutation probability is 0.2, the tournament size is 1/2 of the considered group of individuals and convex recombination is used. The dominance probability has randomly generated values between 0.45 and 0.65 [2]. The error values

		UL	UG	OL	OG	WL	WG
f_1	B	1.49E+04	1.58E+04	8.16E+02	**4.34E+02**	2.18E+04	1.76E+04
	M	6.68E+04	6.53E+04	2.71E+03	**2.05E+03**	5.23E+04	5.95E+04
	S	2.74E+04	2.53E+04	1.44E+03	9.30E+02	1.98E+04	2.01E+04
f_2	B	1.68E+10	2.21E+09	1.30E+07	1.87E+07	1.08E+10	8.80E+09
	M	8.43E+10	8.01E+10	2.27E+08	**1.28E+08**	1.00E+11	1.08E+11
	S	4.23E+10	6.08E+10	2.58E+08	7.55E+07	6.40E+10	5.75E+10
f_3	B	2.13E+01	2.10E+01	2.04E+01	**2.02E+01**	2.14E+01	2.13E+01
	M	2.17E+01	2.17E+01	2.07E+01	**2.07E+01**	2.17E+01	2.17E+01
	S	1.45E-01	2.40E-01	1.40E-01	1.65E-01	2.23E-01	2.41E-01
f_4	B	1.51E+02	1.17E+02	1.79E+01	2.18E+01	1.67E+02	1.41E+02
	M	2.38E+02	2.20E+02	3.46E+01	3.88E+01	2.43E+02	2.48E+02
	S	5.71E+01	4.58E+01	9.36E+00	9.75E+00	4.15E+01	5.92E+01
f_5	B	1.58E+02	2.17E+02	3.89E+01	**3.15E+01**	1.70E+02	2.39E+02
	M	3.68E+02	4.23E+02	6.69E+01	**5.77E+01**	4.28E+02	3.80E+02
	S	1.07E+02	1.28E+02	1.60E+01	1.35E+01	1.28E+02	8.96E+01
f_6	B	1.41E+01	1.43E+01	6.93E+00	6.99E+00	1.42E+01	1.35E+01
	M	2.08E+01	2.05E+01	9.83E+00	**9.44E+00**	1.99E+01	2.07E+01
	S	2.84E+00	2.86E+00	1.31E+00	1.35E+00	3.27E+00	3.15E+00

TABLE 1. Numerical results for the considered DACE strategies: best error values (B), average error values (M), standard deviations (S)

$f(x) - f(x^*)$ where x^* is the real optimum, are presented in Table 1. The best and the average error values have been recorded after 1E+3 function evaluations (FEs), after 25 runs of each algorithm for each function with dimension D=10. The obtained standard deviations are also presented in the table.

The results clearly show an overall better performance of the DACE_OG algorithm compared to the other DACE variants. This is an indication of the benefits generated by a fitness-ranked population topology and an agent-inspired collaboration strategy for selection and recombination. Global search delivers better results compared to the local strategy.

The numerical results obtained using a sorted population structure (DACE_OL, DACE_OG) are clearly superior to those obtained by the DACE variants where the population is distributed in layers without sorting individuals according to fitness (DACE_UL, DACE_UG). Furthermore, the DACE variants having no agent-based behaviour (DACE_WL, DACE_WG) are outperformed by the DACE models where the search process is guided by the interactions between individuals belonging to societies with different strategies.

Similar results have been obtained for 1E+4 FEs and for functions with dimension D=30.

5. Conclusions

A distributed evolutionary asynchronous search model is described and analysed. The main features of the model refer to the geometrical structure of the population guided by the fitness of individuals and the asynchronous search process facilitated by collaboration between agent-inspired subpopulations.

The importance of sorting the population based on fitness, local/global search strategies and society-differentiated behaviours are analysed based on a set of numerical experiments for the optimization of complex functions. Numerical results clearly emphasize the benefits of fitness-ranked population topology and agent-based search integrating local, far and global selection strategies. The collaboration between individuals with various selection/recombination strategies facilitates a benefic adaptive population behaviour in evolutionary models.

References

[1] C. Chira, A. Gog, and D. Dumitrescu. Distribution, collaboration and coevolution in asynchronous search. In Proceedings of the International Symposium on Distributed Computing and Artificial Intelligence, Advances in Soft Computing, pages 596-604. Springer, 2009.

[2] A. Gog, C. Chira, and D. Dumitrescu. Asynchronous evolutionary search: Multi-population collaboration and complex dynamics. In Congress on Evolutionary Computation (CEC 2009), accepted, 2009.

[3] S. Russel, P. Norvig. Artificial Intelligence: A Modern Approach, Prentice Hall, 2002.

[4] K. Tang, X. Yao, P. Suganthan, C. MacNish, Y. Chen, C. Chen, and Z. Yang. Benchmark functions for the CEC'2008 special session and competition on large scale global optimization. Technical Report, USTC, China, 2008.

[(1)] Department of Computer Science, Babes-Bolyai University, Kogalniceanu 1, 400084 Cluj-Napoca, Romania

E-mail address: anca,cchira,ddumitr@cs.ubbcluj.ro

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 103–106

A STEP-BACK SENSITIVE ANT MODEL FOR SOLVING COMPLEX PROBLEMS

CAMELIA CHIRA[(1)], CAMELIA-MIHAELA PINTEA[(1)], AND D. DUMITRESCU[(1)]

Abstract. Ant colony optimization models are investigated with the aim of designing a new ant-based model better suited for real-world complex problems. The proposed *Step-Back Sensitive Ant Model (SB-SAM)* is inspired by the real behaviour of Lasius niger ants including u-turns in the process of selection. A 'step back' is taken by an agent if it reaches a virtual state modulated by various sensitivity levels to the pheromone trails. An effective exploration of the search space is performed particularly by agents having low pheromone sensitivity while the exploitation of intermediary solutions is facilitated by highly-sensitive ants. Numerical experiments carried out for solving the Linear Ordering Problem emphasize a better performance of the proposed SB-SAM algorithm compared to the Ant Colony System (ACS) technique.

1. Introduction

The Ant Colony System (ACS) model [1] relies on the stigmergic interactions between many identical agents (or ants) to find solutions to a given problem. Inspired by the real-world collective behaviour of social insects, Ant Colony Optimization (ACO) algorithms have been successfully applied to a variety of combinatorial optimization problems ranging from quadratic assignment and scheduling to protein folding or vehicle routing. In the ACS model, each ant generates a complete tour (associated to a problem solution) by probabilistically choosing the next node at each path intersection based on the cost and the amount of pheromone on the connecting edge (according to the state transition rule) [1]. Stronger pheromone trails are preferred by ants and the most promising tours build up higher amounts of pheromone in time.

It is proposed to extend the ACS model inducing heterogeneity in the population by endowing each ant with a different level of sensitivity to pheromone trails. Additionally, the extended model takes further inspiration from biology (specifically from the real behaviour of Lasius niger ants) allowing ants to make u-turns and take a step back on the path. The proposed model is called *Step Back Sensitive Ant Model (SB-SAM)* and modulates ant behaviour by defining a virtual state for ants having a certain sensitivity level. The virtual state transition rule avoids a selection of the next

2000 *Mathematics Subject Classification.* 68T20.
Key words and phrases. Ant Colony System, Pheromone Sensitivity, Nature-inspired Metaheuristics.

step from the available nodes forcing the agent to take a 'step back' to the previous node and explore other regions of the search space.

Numerical experiments focus on the Linear Ordering Problem (LOP) - a well-known combinatorial optimization problem seeking a permutation of rows and columns in a given matrix of weights in order to maximize the sum of weights in the upper triangle [2]. The results reported by SB-SAM are better than those of ACS indicating the potential of inducing heterogeneity in ant-based models for better addressing real-world difficult problems where complex behaviour patterns are needed.

2. The Step-Back Sensitive Ant Model

The *Step-Back Sensitive Ant Model (SB-SAM)* extends the ACS model [1] by inducing heterogeneity in the ant population and by allowing ants to make u-turns during the search process. The aim is to enhance the search capabilities of the system particularly for complex solution spaces.

Each agent is endowed with a pheromone sensitivity level denoted by *PSL* (expressed by a real number in the unit interval) which triggers many types of reactions to a changing environment. The variable pheromone sensitivity within the same population of ants can potentially intensify the search (through high PSL values) while in the same time inducing diversity for the exploration of the environment.

In SB-SAM, the probability (used in the ACS model [1]) of choosing the next node from those available is renormalized using the PSL values for each agent in such a way that each potential future transition is associated a lower probability. An agent k selects the next node u from the current node i at time t with the probability $sp_{iu}(t,k)$ given by:

$$sp_{iu}(t,k) = p_{iu}(t,k) \cdot PSL(t,k), \tag{1}$$

where $p_{iu}(t,k)$ represents the ACS transition probability [1] and *PSL(t, k)* is the *PSL* value of agent k at time t.

A *virtual state* can be selected by ant k with probability $1-PSL(t,k)$. The action associated with the virtual decision rule is nature-inspired: it has been observed that the Lasius niger ants include u-turns in the process of selection which have a high impact on the quality of the detected paths [3]. Therefore, the virtual state transition rule specified by SB-SAM makes the ant to take a 'step back' by selecting the previous node. The pheromone trail is locally updated by decreasing the pheromone intensity on the edge connecting the current node with the previous one. The search continues from the new current node until a complete tour is built.

It should be noted that the probability of an agent to take the virtual state decision is inverse proportionally with the agent's sensitivity level: lower PSL means higher chance to take a step back whereas high PSL values favour the application of the ACS-inherited state transition rule.

3. Numerical Experiments

The proposed SB-SAM algorithm is compared to the ACS model in solving several LOP instances.

LOP is a well-known NP-hard problem having a wide variety of real-world applications including triangulation of input-output matrices, single-machine scheduling and aggregation of individual preferences within different contexts [2]. LOP can be formulated in graph theory terms of searching for an acyclic tournament having the maximum sum of arc weights in a complete weighted graph. An acyclic tournament corresponds to a permutation (or an ordering) of the graph vertices.

A LOP solution is a list of n vertices constructed in a step by step manner by each ant based on the transition rules specified by the ant model. Both ACS and SB-SAM algorithms are initialized with a greedy solution [4] obtained based on a 2-exchange neighbourhood search (solutions obtained by permuting two positions in the ordering). In the SB-SAM approach, the PSL value for each agent is randomly generated each cycle using an uniform distribution.

Numerical experiments use the 49 LOP instances available in the well-known LOLIB [5] real-world data library. LOLIB provides the known optimum solution for each problem instance. The deviation of the obtained solution from the optimum solution is computed for each problem instance. The results are recorded based on the average over five runs of the algorithm with 5000 iterations each.

To compare the performance of the two ant-based models the average deviation is analysed for each problem size from LOLIB: 44, 50, 56 and 60. A better overall performance of SB-SAM is clearly supported by the obtained results (the deviation of SB-SAM solutions from the optimum is generally lower compared to ACS solutions while for a small number of problem instances ACS obtains a slightly better average deviation).

Table 1 presents the numerical results obtained for ACS and SB-SAM for instances of size 50. The problem instance and the known optimal solution are given in the first two columns of Table 1. The other two columns present the average deviation of ACS and SB-SAM obtained solutions from the known optimum. It can be observed that SB-SAM obtains a lower deviation for all problems in Table 1 indicating that the solution obtained by SB-SAM is closer to the optimum solution compared to ACS. Similar results have been obtained for the other problem instances considered (not given here due to paper length restrictions).

The proposed SB-SAM model obtains solutions of a higher-quality compared to the standard ACS model emphasizing the benefits of heterogeneous agent behaviour (induced by the PSL distribution in the population) and the potential of supplementary search space exploration generated by the step-back virtual decision rule.

4. Conclusions

The SB-SAM technique is an ant-based model able to facilitate search diversification by allowing some ants to make u-turns and return to the previous node under certain conditions. Ants are endowed with various sensitivity levels to pheromone trails generating different reactions to the same environment. Agents with low pheromone

TABLE 1. Average deviation from optimal solution obtained by ACS and SB-SAM for LOLIB instances of size 50

Problem Instance	Optimal Solution	ACS Avg.Dev.	SB-SAM Avg.Dev.
50-be75tot	1127387	0.0034	**0.0033**
50-be75oi	118159	0.0006	**0.0001**
50-be75np	790966	0.0004	**0.0003**
50-be75eec	264940	0.0024	**0.0018**

sensitivity tend to select a virtual state transition taking a 'step back' to the previous node and exploring other search regions. Numerical experiments indicate a better performance of SB-SAM compared to the ACS technique in solving various instances of the linear ordering combinatorial optimization problem.

REFERENCES

[1] M. Dorigo, L.M. Gambardella. Ant Colony System: A cooperative learning approach to the traveling salesman problem, IEEE Trans. on Systems, Man, and Cybernetics, 26, 29-41,1996.
[2] S. Chanas, P. Kobylanski. A new heuristic algorithm solving the linear ordering problem, Computational Optimization and Applications, 6,191-205, 1996.
[3] R. Beckers, J.L. Deneuborg, S. Goss. Trails and U-turns in the Selection of a Path of the Ant Lasius Niger, J. theor. Biol., 159, 397-415, 1992.
[4] M.G.C. Resende, C.C. Ribeiro. Greedy randomized adaptive search procedures: Advances and applications, Handbook of Metaheuristics, 2nd Edition, Springer, 2008.
[5] http://www.iwr.uni-heidelberg.de/groups/comopt/software/LOLIB/, LOLIB site, at Heidelberg University, Germany.

[1] DEPARTMENT OF COMPUTER SCIENCE, BABES-BOLYAI UNIVERSITY, 400084 CLUJ-NAPOCA, ROMANIA

E-mail address: {cchira, cmpintea, ddumitr}@cs.ubbcluj.ro

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 107–110

IMPROVING DEFINITION ALIGNMENT BY SVM WITH A KERNEL OF KERNELS

LAURA DIOŞAN[(1)] AND ALEXANDRINA ROGOZAN[(2)], JEAN-PIERRE PECUCHET[(2)]

Abstract. The alignment between a specialized terminology used by the librarians to index concepts and a general vocabulary employed by a neophyte user in order to retrieve documents on Internet, will certainly improve the performances of the information retrieval process. We develop a medical terminology alignment by an SVM classifier with an optimised kernel trained on a compact, but relevant representation of such definition pairs by several similarity measures and the length of definitions. The results obtained on the test set show the relevance of our approach, as the F-measure reaches 88%. However, this conclusion should be validated on larger corpora.

1. Introduction

The need for terminology integration has been widely recognized in the medical area leading to a number of efforts for defining standardized terminologies. It is, however, also acknowledged by the literature, that the creation of a single universal terminology for the medical domain is neither possible nor beneficial, because different tasks and viewpoints require different conceptual choices [2].

This situation demands for a weak notion of integration, also referred to as alignment in order to be able to exchange information between different communities. In fact, the commonalities of two different terminologies have to be found in order to facilitate interoperability between computer systems that are based on these two terminologies. In this way, the frontiers between general language and specialist one could be linked.

The goal of our work is to enrich the information retrieval system with a set of links that correspond to an alignment of two concepts, which allow a better exploitation of specialised terminologies and electronic dictionaries in order to benefit from the advantages of their strengths. A non-expert user would therefore access documents indexed through the concepts of a professional dictionary if they are correlated by semantic links to a general dictionary. An important idea is to look for the terms indexed by an expert by using a non-specialized vocabulary and vice versa. In this context, one of the most important tasks is to achieve an automatic alignment of concepts. The main aim is actually to find a mapping between different sequences of

2000 *Mathematics Subject Classification.* 68T05,91E45.
Key words and phrases. Alignment, SVM, Kernel of kernels, Genetic Programming.

words, but of the same meaning, in our case the meaning of a concept being represented by its definition(s) from one or more dictionaries (i.e. to associate definitions from different dictionaries that correspond to the same or to similar concept). Therefore, the concept alignment is actually viewed in terms of definition alignment.

This paper is structured as follows: Section 2 details the characteristics of the corpora and of the alignment model, which is analysed (through several numerical experiments) in the next section. Finally, Section 4 concludes the paper.

2. Corpora and alignment model

To our knowledge, only the problem of aligning sentences from parallel bilingual corpora has been intensively studied for automated translation. While much of the research has focused on the unsupervised models [3], a number of supervised discriminatory approaches have been recently proposed for automatic alignment [1].

The growing number of resources to be indexed in the catalogue of online health resources in French (CISMeF) calls for creating strategies involving automatic indexing tools while maintaining the catalogue's high indexing quality standards. To this end, Neveol et al. [4] developed several automatic tools that retrieve MeSH descriptors from documents titles.

One of the most important tasks is to achieve an automatically alignment of concepts. In our case, the meaning of a concept is represented by its definitions from more dictionaries. This alignment of definitions, which is the main goal of the VODEL project as well, has certainly to improve the fusion between a specialized terminology and a general vocabulary employed by a neophyte user in order to retrieve documents on Internet. Therefore, the concept alignment is actually viewed in terms of definition alignment.

In order to automatically perform this alignment, more definitions are considered from different dictionaries. Pairs of two definitions (that can define the same concept – and in this case we deal with two aligned definitions – or different concepts – the definitions being not aligned in this case) are formed. Thus, the alignment problem could be considered as a binary classification problem: the inputs of the classifier are the pairs (of two definitions) and the outputs are the labels "aligned" or "not aligned" corresponding to each pair.

The corpus has been realised during the VODEL project by a team of linguistic specialists (G. Lortal, I. Bou Salem and M. Wang). They have chosen two specialised dictionaries (MeSH and VIDAL) and two general dictioanries (LDI and WIKIPEDIA). Several preliminary treatments of these definitions have been perfomed, since the literature shows that a purely statistical approach on the plain text provides weak results for automatic text understanding. This linguistic processing refers to: segmentation, stop-word filter, lemmatisation and syntactic labelling. Since the VODEL project was interested in how different syntactic levels influence the performances of the alignment process, the definitions have been considered at different syntactic levels: nouns, nouns and adjective, nouns, adjectives and verbs.

We use several measures of similarity between two structures: the *Matching*, the *Dice*, the *Jaccard*, the *Overlap* and the *Cosine* measure. By working only with a

representation based on these measures, instead of a classical one, the models we develop are able to map the initial vectors (based on a bag of word approach) into a space of reduced dimension, where the computation effort is smaller.

The alignment is considered as a classification problem where each input is represented by the similarity between two definitions and the label associated to that couple of definitions (aligned or not aligned) represents the output. An SVM algorithm is actually used to perform this classification-alignment.

In order to classify the definition couples, the SVM algorithm uses the above representations and a kernel function. The parameters of the SVM model (the penalty for miss-classification C and the kernel parameters) are optimized on the validation set by using the model proposed in [5].

This model was aimed to automatically design a complex multiple kernel (called Kernel of kernels - KoK) and to optimise its parameters by evolutionary means. In order to achieve this purpose, a hybrid model that combines a Genetic Programming (GP) algorithm and a kernel-based Support Vector Machine (SVM) classifier was proposed. Each GP chromosome is a tree that encodes the mathematical expression of a KoK function. The purpose of this model was to find the best KoK function and to optimise its parameters, but also to adapt the regularisation kernel parameter C. These three objectives are achieved simultaneously because each GP chromosome encodes the expression of a KoK and its parameters. The GP-kernel is involved into a standard SVM algorithm to be trained in order to solve a particular classification problem. After an iterative process which runs more generations, an optimal evolved KoK is provided. The proposed combination of kernels could be learnt from thousands of examples while combining hundreds of kernels within reasonable time. Thus, the SVM classifier is automatically adapted to the problem, actually the alignment of definitions.

3. Experimental validation and discussions

Several numerical experiments are performed by using our model for six different combinations of dictionaries and for different syntactic levels. Two kernel types are considered: a well-known RBF kernel and an evolved kernel of kernels. Several performance measures, borrowed from the information retrieval domain, are utilised in order to evaluate the quality of the automatic alignments: the precision of alignments, the recall of alignments and the F-measure.

The results on the test set disjoint from that training one show the relevance of our original approach based on the concatenation of the similarity measures, since the F-measure reached 87.99%. In the case of using an RBF kernel, the model based on learning from nominal groups like NA (Nouns-Adjectives) leads to better performance. In the case of an evolved kernel of kernels the best results are obtained for the MEMO *vs.* MeSH combination at the syntactic levels of nouns. In some cases our evolved kernel of kernels improves the performance of the classifier.

4. Conclusions and remarks

In this paper we presented our model for the automatic alignment of definitions taken from general and specialised dictionaries. The definitions have been considered at three syntactic levels and the influence of each level has been analysed. The best performances are obtained by using the SVM algorithm and the evolved kernel of kernels, since the classifier (in fact its kernel function and the hyper-parameters) is better adapted to the alignment task to be solved.

Further work will be focused on: considering a representation of definitions enriched by semantic and lexical extensions (synonyms, hyponyms, and antonyms) and on developing of an alignment model based on an SVM algorithm with a specialised multiple kernel (this specialisation could be considered in terms of combination of more kernels for text processing (*e.g.* string kernels)).

References

[1] A. Ceausu, D. Stefanescu, and D. Tufis. Acquis communautaire sentence alignment using SVMs. In *The 5th LREC Conference*, pages 2134–2137, 2006.

[2] A. Gangemi et al. An overview of the ONIONS project: Applying ontologies to the integration of medical terminologies. *Data Knowl. Eng*, 31(2):183–220, 1999.

[3] P. F. Brown et al. The mathematic of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, 1994.

[4] A. Névéol, S. E. Shooshan, and V. Claveau. Automatic inference of indexing rules for MEDLINE. *BMC BioInformatics*, page in press, 2008.

[5] L. **Dioşan**, A. Rogozan, and J.-P. Pécuchet. Optimising multiple kernels for SVM by GP. In J. van Hemert and C. Cotta, editors, *EvoCOP 2008*, volume 4972 of *LNCS*, pages 230–241. Springer, 2008.

[1] Babeş Bolyai University, Cluj-Napoca, Romania
E-mail address: lauras@cs.ubbcluj.ro

[2] INSA, Rouen, France
E-mail address: {arogozan,pecuchet}@insa-rouen.fr

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 111–114

EQUILIBRIA DETECTION IN ELECTRICITY MARKET GAMES

D. DUMITRESCU[1], RODICA IOANA LUNG[2], AND TUDOR DAN MIHOC[3]

Abstract. A general technique for finding equilibria in finite non cooperative games is applied to compute Nash equilibria of an electricity market. The approach is based on a characterization of equilibrium using a generative relation. This relation induces an appropriate domination concept. A population of strategies is evolved according to a domination-based ranking. The process converges towards a very good approximation of the game equilibrium.

1. Introduction

Detecting Nash equilibrium is a fundamental computational problem in finite noncooperative games [1]. In a pure strategy Nash equilibrium each decision-maker plays a pure, non necessarily dominant strategy, that is the best response to the strategies of other players.

An evolutionary technique for detecting equilibria is used. Equilibrium is characterized by a binary relation on the game strategies, called generative relation. The relation induces an appropriate domination concept. Game equilibrium is described as the set of non dominated strategies with respect to the generative relation. For equilibrium detection a population of strategies is evolved according to a fitness concept given by domination-based ranking. The process converges towards the game equilibrium.

This approach is used to compute efficiently the Nash equilibrium for an electrical spot market modeled as a noncooperative game [2].

2. Basic notions

Some basic notions related to game theory are presented. A finite strategic game is defined as a system $\Gamma = ((N, S_i, u_i), i = 1, n)$ where:

- $N = \{1, ..., n\}$ is a set of n players;
- for each player $i \in N$, S_i represents the set of actions (pure strategies) available to her, $S_i = \{s_{i_1}, s_{i_2}, ..., s_{i_{m_i}}\}$;
- $S = S_1 \times S_2 \times ... \times S_N$ is the set of all possible situations of the game;

2000 *Mathematics Subject Classification.* 91A80,68T01.

Key words and phrases. Nash equilibrium, generative relation for equilibria.

[3]"Investeşte în oameni!" doctorand bursier în "Proiect cofinanţat din Fondul Social European prin Programul Operaţional Sectorial Dezvoltarea Resurselor Umane 2007-2013" .

- an element of S is a strategy profile (or strategy) of the game;
- for each player $i \in N$, $u_i : S \to R$ represents the payoff function.

Let s^* be a strategy profile. Denote by (s_{i_j}, s^*_{-i}) the strategy profile obtained from s^* by replacing the strategy of player i by s_{i_j} i.e.

$$(s_{i_j}, s^*_{-i}) = (s^*_i, s^*_2, ..., s^*_{i-1}, s_{i_j}, s^*_{i+1}, ..., s^*_n).$$

S_{-i} denotes a strategy profile of every player except i.

Definition Profile strategy s^* is a Nash equilibrium [4] if the inequality $u_i(s^*) \geq u_i(s_{i_j}, s^*_{-i})$ holds for every action s_{i_j} of player i, $s_{i_j} \in S_i$.

Let $k(s', s'')$ denotes the number of individual strategies from s' which replaced in s'' give better payoff for the corresponding player.

$$k(s'', s') = card\{i \in \{1, ..., n\} | u_i(s'_i, s''_{-i}) \geq u_i(s''), s'_i \neq s''_i\}$$

Otherwise stated $k(s'', s')$ is the number of players benefiting by switching from s'' to s' and measures the sensitivity of s'' with respect to perturbations supplied from s'. The lower sensitivity, the higher is the stability of s'' with respect to s'.

We may use

$$m(s'', s') = n - k(s'', s')$$

as a measure for the relative quality of s'' with respect to s'.

Let us consider a relation R_N on $S \times S$: $(s', s'') \in R_N$ if and only if s' is better than s'' with respect to m, i.e. $m(s', s'') > m(s'', s')$. Therefore $(s', s'') \in R_N$ if and only if $k(s', s'') < k(s'', s')$.

Proposition 1 [5] R_N is the generative relation of the Nash equilibrium, i.e. non-dominated strategies with respect to R_N are the Nash equilibria of the game.

3. Evolutionary equilibria detection

Let R be the generative relation for a the specific equilibrium E.

A sequence of approximations of equilibria set E may be constructed using selection methods based on generative relation R and variation operators.

A population of strategies is evolved. The initial population is randomly generated. Strategy population at iteration t may be regarded as the current equilibrium approximation. Subsequent application of such operators (like the simulated binary crossover (SBX) [3] and real polynomial mutation [3]) is guided by a specific selection operator induced by the generative relation.

Selection for survival can be done by using a procedure based on the same selection operator or another one, also correlated to the generative relation. In this way successive populations produce new approximations of the equilibrium front.

The previous approach can be summarized in a technique called Relational Evolutionary Equilibria Detection (REED) as described below.

REED algorithm

S1. Set $t = 0$;

S2. Randomly initialize a population $P(0)$ of strategies;

S3. Binary tournament selection and recombination using the simulated binary crossover (SBX) operator for $P(t) \to Q$;

	Q_{max} [MW]	c_{min} [\$/h]	c_{max} [\$/h]
G1	100	800	2500
G2	200	1300	6000
G3	300	1800	9000

TABLE 1. Generators data

	G1	G2	G3
Avg.	65.6	87.2	114.03
St. dev.	0.02	0.12	0.43

TABLE 2. Average of detected equilibrium approximation and standard deviation in the final population of 100 strategies after 100 generations

S4. Mutation on Q using real polynomial mutation $\rightarrow P$;
S5. Compute the rank of each population member in $P(t) \cup P$ with respect to the generative relation. Order by rank $(P(t) \cup P)$;
S6. Rank based selection for survival $\rightarrow$ P (t + 1);
S7. Repeat steps S3 - S6 until the maximum generation number is reached.

4. Numerical experiments

We asses in this section our approach to compute the Nash equilibria on an electricity market. The spot market consists in three generators G1, G2 and G3. All generators have a limited production capacity $Q_{i\ max}$ and a production cost

$$CM_i = \frac{Q_i^2}{Q_{i\ max}^2} \times (c_{i\ max} - c_{i\ min}) + c_{i\ min}$$

where Q_i is the amount of power the generator i is scheduled to produce, $c_{i\ max}$ is the production cost when the generator i works at full capacity and $c_{i\ min}$ is the generator i maintaining costs per hour. The values for each generator are defined in Table 1. We consider a price spot market with a price cap of 50\$/MWh. The equation

$$Pr_{market} = (1 - \frac{Q_{tot}}{Q_{1\ max} + Q_{2\ max} + Q_{3\ max}}) * 50$$

gives the price of the MWh when a total amount of energy Q_{tot} is produced. The profit u_i for generator i is computed using the following expression:

$$u_i = Q_i * Pr_{market} - CM_i.$$

The strategy profile $(s_1, s_2, s_3) \in S$ consists in the amount of power produced by each generator. A very good approximation of Nash equilibrium has been detected without difficulty for the game. In Table 2 we have the average and the standard deviation of the final population.

5. Conclusions

An evolutionary technique for detecting approximations of non cooperative game equilibria has been used and evaluated. A game that simulates a spot market with three players is used to exemplify the detection Nash equilibrium. The proposed method, based on generative relations, allows to compute with relative small resources the equilibrium. A qualitative study of the detected approximation also denotes the efficiency of method.

References

[1] Bade, S., Haeringer, G., Renou, L.: *More strategies, more Nash equilibria*, Working Paper 2004-15, School of Economics University of Adelaide University, 2004.

[2] Beck, E., V., Cherkaoui, R., Minoia, A., Ernst, D.: *Nash equilibrium as the minimum of a function. Application to electricity markets with large number of actors*, 7th IEEE Power Engineering Society (PowerTech 2007), page 1–5 - jul 2007

[3] Deb, K., Agrawal, S., Pratab, A., Meyarivan, T.: *A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II*, IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, VOL. 6, NO. 2, APRIL 2002

[4] Nash, J.,F.: *Non-cooperative games*, Annals of Mathematics, 54:286-295, 1951.

[5] Lung,R.,I., Dumitrescu, D.: *Computing Nash Equilibria by Means of Evolutionary Computation*, Int. J. of Computers, Communications & Control, 2008, 364-368

[6] Maskin, E.: *The theory of implementation in Nash equilibrium:A survey*, in: L. Hurwicz, D. Schmeidler and H. Sonnenschein, eds., Social Goals and Social Organization (Cambridge University Press), 1985,173-204

[7] Osborne, M.: *An Introduction to Game Theory*, Oxford University Press, 2004

[1] Babeş Bolyai University Cluj Napoca
E-mail address: ddumitr@cs.ubbcluj.ro

[2] Babeş Bolyai University Cluj Napoca
E-mail address: rodica.lung@econ.ubbcluj.ro

[3] Babeş Bolyai University Cluj Napoca
E-mail address: tudormihoc@yahoo.com

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 115–118

UNIVERSICS – A STRUCTURAL FRAMEWORK FOR KNOWLEDGE REPRESENTATION

IOACHIM DRUGUS[(1)]

Abstract. A "theory of universes", or *Universics*, was developed as a framework for integration of disciplines via their *universes of discourse*. The formal apparatus of Universics is obtained by a revision of set theory, motivated by the necessity to take into account the agents processing sets. This revision lead to entities of a new kind said to be the *universes*. The content of mind is regarded as a universe built according the *atom-aggregation-association*, or *A3*, approach, which in [1] is placed in the foundation of the workings of mind. By "inverse engineering", the "form"of any universe is declared as amenable of description via the data model of the A3 approach. The meaning is assigned to universes via (homo)morphisms of universes. A universe with meaning is said to be a *world* - a treatment which complies with the usage of this term in Semantic Web. The integration of knowledge of the worlds of different disciplines can be achieved via application of *functors* of category theory. At *SemanticSoft, Inc.*, a software system for knowledge representation based on the data model of *Universics* is in the process of development.

The strength of the Leibniz' metaphor of *possible worlds* was demonstrated by its conceptual service to the development of at least two frameworks - the Kripke models [2,3] for modelling non-classical logics, and the *forcing* method in set theory, which allowed to solve one of the most complex Hilbert problems, the "continuum hypothesis" problem N1. The aim of this research is to place the Leibniz' metaphor of 'possible worlds' in the basis of a formal *holistic* discipline which would use a "part-whole" relationship by involving also the largest "whole", the Universe. I will refer to this discipline as *Universics*. In Universics I will treat anything which has a *form* as a *universe* and any universe, the entities of which are meaningful - as a *world*. This paper is focused on the data model of Universics, while the operational aspect of this discipline will be described in a chapter of the book [4].

1. Worlds and Universes

The dictionaries assign different meaning to the two words, "World" and "Universe" - "World" is the name of the planet Earth seen from the human point of *view*, and "Universe"- denotes "everything that constitutes reality". I will use these two words as plurals, thus, changing their status from names of individuals to names of

2000 *Mathematics Subject Classification.* 03C99.
Key words and phrases. knowledge representation, universics, modelling.

classes. Also, based on the distinction emphasized by dictionaries, I will consider, that a *world* is a result of a *view* of an intelligent agent upon a *universe*. There can exist many such views upon just one universe, *the Universe* (which can be said to be "worldviews") and this is responsible for the multiplicity of worlds. Any view can be treated as limited to a part of the Universe, which I will say to be a *body* and which I will treat also as a universe.

I will also treat a view upon a universe V as a constraint on the possible interpretations of the universe. This treatment of view can be regarded as a generalization of the notion of *interpretation* treated in Semantic Web as a *closed world* interpretation. A view can be treated as an *open world* interpretation or, simply, *open interpretation*. This intuition leads to the category of universes and views (open interpretations), in the sense of category theory. Also, the *intentionality* reasons of *phenomenology* demand to regard a view as oriented towards an object, and make abstraction from the subject. Therefore, I define a world W, treated as obtained in result of a view, as an ordered pair (v, V) where v is a *morphism* from a universe U, *subject* of view, to a universe V, *object* of view.

2. The A3 approach

To further formalize the notions introduced above, we need to specify what is a *form*, or a "structure", of a *universe*. Here, I used "structure" between quotation marks, because this word is reserved for "structures within" a universe, and I will accept, that alongside structures, a universe can contain also *atoms* which, by definition, are "non-structures". I will regard, the structures to be built via a construction process, which can proceed from atoms, but which can also create an atom out of a structure, by encapsulating it, i.e., by limiting the view to only the identity of a structure. Since a form can be structured into many structures, I regard *form* to be a more fundamental notion than "structure".

In formalization of Universics, I am making use of the *atom-aggregation-association* approach, or *A3* approach, to brain informatics as outlined in [1], which I also regard as a method native to Universics. The A3 approach is used to define the form of a body of knowledge, and via a procedure which I am saying to be "inverse engineering," I will consider any universe to be "in the image" of its representation in mind and I will assign to it the same form as its image in mind.

According the A3 approach (formulated here informally), all the logical operations can be reduced to three logical operations: (1) *Atomification* - associating with a *structure* an entity of type *atom* said to be identity of the structure (an operation which can be ascribed to the bridge between cerebral hemispheres, called corpus callosum), (2) *Aggregation* - creating a set and associating with it an atom of type *aggregation* said to be identity of the set (operation ascribed to right cerebral hemisphere, for most people) (3) *Association* - creating an ordered pair and associating with it an atom of type *association* (operation ascribed to left cerebral hemisphere, for most people)

An atom, set or an ordered pair may obtain many identities, so that the names of these entities may be modified by the prefix *multi*: multi-atoms said to be *individuals*

to keep compliance with of Semantic Web, multi-sets said to be *aggregations*, and multi-ordered-pairs, said to be *association triples*. The universe of such entities is quite different from the universe of set theory with the "extensionality axiom". The axiomatization of this domain is planned to be done in [4]. I will say the result of multiple application of these operations to be a *body*. If in construction of a body no atom is re-used to serve as an identity, i.e., if in each application of one of the three operations a new identity is created, so that the identity *identifies* the entity with which it is associated, then the body will be said to be a *universe*. We can regard the application of each of the three operations as independent on each other and, similar to physics, will assign a "dimensionality" to each body or universe, said to be a "universal dimensionality" - scale (the dimensionality of atoms), space (dimensionality of aggregations), time (dimensionality of associations) (see also [1]). Based on these informal considerations, I will introduce formally the basic notions of Universics in the next section.

3. The Category of Universes

A *body* B is a tuple (A, A_1, A_2, A_3), where A is a class said to be *foundation* of, and its elements – *entities* of, the body B, and also

(1) A_1 is a 1-ary relationship of "being an atom", or *atomicity property*,
(2) A_2 is a 2-ary *aggregation relationship* such that for any (x, i) in A_2 , i is an atom and is said to be identity of the set $\{x \in A | (x, i) \in A_2\}$,
(3) A_3 is a 3-ary *association relationship* such that for any (x, i, y) in A_3, the entity i is an atom said to be identity of the ordered pair (x, y).

An argument k of a n-ary relationship R is said to be its *primary key*, if for any x_1,..., x_n, y_1,..., y_n such that $R(x_1, ..., x_n)$ and $R(y_1, ..., y_n)$, if $x_k = y_k$ then

$$(x_1, ..., x_n) = (y_1, ..., y_n)$$

Finally, we can define our central notion. A *universe* is a body, such that the identities of its atoms, sets and ordered pairs are primary keys.

A morphism of a body $B = (A, A_1, A_2, A_3)$ into a body $C = (A', A'_1, A'_2, A'_3)$ is a triple (B, h, C), where h is a function from A to A', such that h preserves each of the relationships A_n, i.e., for n=1,2,3, and a tuple $(x_1, ...x_n)$ in A_n, $h(x_1, ...x_n) = (h(x_1), ...h(x_n))$. This is a very compact definition which allows to generalize the notion of open function from topology, homomorphism from algebra, and several other notions, but it also introduces the notion of preservation of atomicity, i.e. - preservation of discreteness (vs continuality) of a universe. To discover the behavior of such morphisms in each of the three universal dimensions, this definition needs to be examined in detail for each value of n.

A *body in a universe* is the image of a morphism in a universe. Various notions like *motion* of bodies and other notions usually attributed to physics can be introduced, but since we are focused on IT, I will give below only some examples of universes and bodies in the IT domain.

4. Applications of Universics

Examples of universes are the universes of discourse of axiomatic set theories. In such universes, due to the "extensionality axiom", each entity has exactly one identity. Arbitrary universes as defined in this paper contain entities with multiple identities which allows to treat named sets as objects of discourse. Additionally, due to the atomification operation which allows to encapsulate a structure into an atom, such universes allow to model the organization of matter in levels of different depth as in physics: elementary particles, atoms, planets and stars, galaxies, or the organization of the world of biology into bio-molecules, cells, organisms, populations. To distinguish between the atoms in physics and the atoms in a universe as we defined this notion, we will say our atoms to be "logical atoms".

An important class of examples of universes are the ontologies of Semantic Web, which can be treated as *normalized* universes - i.e. universes built up in a uniform manner. Namely, the ontologies are constructed by first applying the association operation and obtaining triples, then aggregating many times the triples in different structures. Such a treatment extracting the pure structure of an ontology helps to find correlations between different languages of Semantic Web.

References

[1] Drugus, I. A. Wholebrain approach to the Web. In: Proceedings of the "Web Intelligence – Intelligent Agent Technology Conference", Silicon Valley, Nov 2-5, 2007.
[2] Kripke, S. A Completeness Theorem in Modal Logic, Journal of Symbolic Logic 24(1):1–14.1959.
[3] Kripke, S. Semantical Considerations on Modal Logic, Acta Philosophica Fennica 16:83–94, 1963.
[4] Web Intelligence and Intelligent Agents, ISBN978-953-7619-X-X, to appear in September, 2009.

[1] Dimo 31/1 Apt. 117, Chisinau, MD-2045, Republic of Moldova
E-mail address: ioachim.drugus@semanticsoft.net

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 119–122

KNOWLEDGE ACQUISITION FROM HISTORICAL DOCUMENTS

IOAN SALOMIE[(1)], MIHAELA DINSOREANU, CRISTINA POP, AND SORIN SUCIU

Abstract. In this paper we present an intelligent system that performs knowledge acquisition from historical documents. Through knowledge acquisition, the content of documents is semantically annotated and the domain ontology is enriched. The paper describes the analysis of the document corpus and the steps for creating the core of the domain ontology as prerequisites of the knowledge acquisition process. Lexical annotation and knowledge extraction are based on domain-dependent components that include rules for extracting the relevant information from the document's content and knowledge acquisition rules, for mapping the extracted information to ontology concepts and properties. Our work was validated on documents addressing the medieval history of Transylvania, proving to be an efficient approach to knowledge retrieval as response to ontologically-guided agent queries.

1. Introduction and Related Work

Digital repositories have become necessary for preserving the content of historical and archival documents. Access to document digital copies can be granted to a wider range of users without the risk of deteriorating the originals. In order to handle effectively a great number of digital documents, it is useful to provide mechanisms to automatically process, manage and retrieve relevant information. Knowledge can be discovered by applying reasoning, learning and data mining techniques on digital documents repositories. The heterogeneity of such natural language documents makes them hard to be processed by machines. This problem can be solved by designing methods for knowledge extraction, creating knowledge representation models and mapping the extracted knowledge to the models.

The objective of our work consists in developing an intelligent system which creates a digital repository of semantically annotated documents targeting machine reasoning and learning based on content data. The obtained knowledge can be used to provide researchers and historians a means to obtain relevant results to

2000 *Mathematics Subject Classification.* 68P20, 68Q55, 68T30, 68U35, 68M14.
Key words and phrases. ontology core, historical documents, semantic annotation, knowledge acquisition, knowledge retrieval.

their queries. We adopted Semantic Web techniques for knowledge capturing, representation and processing. As a result, our system adds a layer of machine-processable semantics over the content of documents by using a domain ontology. Our case studies focused on processing the documents available in the Cluj County National Archives [1] regarding the medieval history of Transylvania.

The system is organized on three interacting processing layers: *(i)* the raw data acquisition and representation layer, *(ii)* the knowledge acquisition layer - enriches the system knowledge by annotating the raw documents with entities from the domain ontology and *(iii)* the knowledge processing and retrieval layer - retrieves relevant documents and knowledge as response to ontologically-guided agent queries. The system follows two main workflows, namely knowledge acquisition on one hand and knowledge processing and retrieval on the other hand. In this paper we detail the knowledge acquisition workflow.

Representative approaches to knowledge acquisition include *(i)* OntoPop, which proposes a solution for performing semantic annotation of documents and ontology population in a single step by using knowledge acquisition rules [3], *(ii)* Ontea, a semi-automatic annotation and information retrieval technique which uses regular expression patterns, lemmatization methods and indexing mechanisms on documents in English and Slovak [5] and *(iii)* SOBA, which builds a soccer knowledge base from Web pages by lexically annotating their content, extracting the relevant information and mapping the annotated entities to ontology elements [4].

2. Knowledge Acquisition Prerequisites

Document corpus. We created our corpus by pre-processing a set of original archival documents capturing historical facts of medieval Transylvania. The historical evolution of Transylvania is the main source of documents heterogeneity, mainly determined by *(i)* the documents language (Latin, Hungarian, German and Romanian), *(ii)* the institution that issued the document (e.g. royal, local or religious authorities), *(iii)* the writing embellishments that decorate the documents. These characteristics lead to difficulties in automatic document processing and therefore we decided to use as corpus the document summaries created by archivists, accompanied by their technical details (e.g. date of issue, archival fund).

Core of the domain ontology. Our system uses a historical domain ontology in the processes of knowledge acquisition, document annotation and semantic querying. The domain ontology is developed in two stages: *(i)* a manual iterative process which creates the core of the domain ontology after a thorough domain analysis and *(ii)* an automatic ontology population process with knowledge extracted from documents. In designing the core of the ontology we studied the medieval history of Transylvania and a large set of corpus documents together with historians and archive experts. This way we have identified the most relevant concepts (e.g. places, persons, dates and events) and relations.

3. Knowledge Acquisition Workflow

The Knowledge Acquisition Layer performs information extraction and document annotation, having as objectives to *(1)* populate and extend the domain ontology with domain-specific information retrieved from the processed documents corpus and *(2)* to annotate the documents with domain ontology entities. Our solution is the domain-independent *Knowledge Acquisition Pipeline* (see Figure 1). The pipeline is inspired by OntoPop [3] and introduces three new processing steps: *(i)* technical data extraction, *(ii)* synonyms population and *(iii)* homonym identification. Specializing the pipeline requires modeling the addressed domain as a set of domain-specific resources: *(i)* the rules to extract the technical data, *(ii)* the pattern-matching rules for lexical annotation, *(iii)* the domain ontology, *(iv)* the knowledge acquisition mapping rules for semantic annotation and ontology population, and *(v)* the thesaurus and dictionary for synonyms. In the following we briefly present the processes in the Knowledge Acquisition Pipeline.

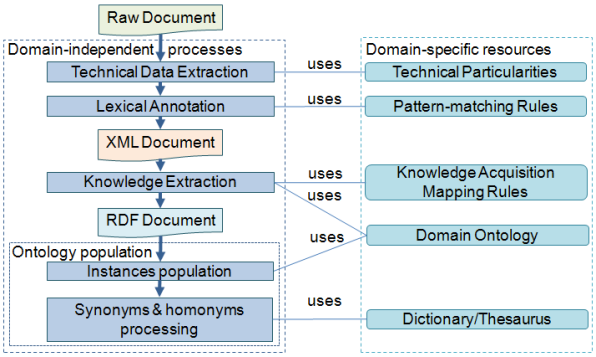


Figure 1. Knowledge Acquisition Pipeline

Technical Data Extraction processes the metadata of the raw documents. The technical data is domain and organization-specific and is created by a domain expert. We use the technical data for determining thresholds involved in reasoning.

Through *Lexical Annotation* the relevant information from the documents' content is extracted and lexically annotated using GATE [6] with a set of pattern-matching rules represented as JAPE grammars [6]. The identified lexical elements are structured as a hierarchy and stored in an XML file.

Knowledge Extraction relies on a set of knowledge acquisition mapping rules for *(1)* semantic annotation of documents using ontological concepts and *(2)* ontology population. Knowledge acquisition mapping rules define *(i)* how to associate ontology concepts to lexical annotations and *(ii)* actions to be executed for populating the domain ontology with new instances from the documents' content. A RDF file containing the semantic annotations of the raw document is also created.

Ontology Population adds to the domain ontology the new instances discovered in the knowledge extraction process, together with their synonyms and properties. We identify homonyms by using a distance function which evaluates the homonymous relationship between a candidate instance and an already stored instance. Ontology population triggers actions which infer new knowledge and maintain the ontology consistency by using SWRL rules and the JESS rule engine [2].

4. Conclusions and Future Work

We presented a solution to knowledge acquisition from historical documents by applying Semantic Web techniques. Knowledge acquisition is organized as a pipeline of processing components that use domain-specific resources enabling technical data extraction, lexical annotation, knowledge extraction and ontology population. The resulting knowledge is composed of the domain ontology populated with instances and relations, together with the RDF files containing the semantic annotations of each processed document. The knowledge is used for information and document retrieval in answer to ontology-guided agent queries.

As for future work proposals, we intend to enhance the domain-specific resources to process multilingual terms. Such an enhancement would allow our system to address a broader set of documents about medieval Transylvania, not only those written in Romanian but also those written in German and Hungarian.

References

[1] Ccna, cluj county national archives, http://www.clujnapoca.ro/arhivelenationale/.
[2] Jess - the rule engine for the java platform, version 7.1, http://www.jessrules.com/jess/docs/jess71.pdf.
[3] R F Amardeilh. Ontopop or how to annotate documents and populate ontologies from texts. In *Proceedings of the ESWC 2006 Workshop on Mastering the Gap: From Information Extraction to Semantic Representation, CEUR Workshop Proceedings*, ISSN 1613-0073, Budva, Montenegro, 2006.
[4] Cimiano P. Racioppa S. Buitelaar, P. and M. Siegel. Ontology-based information extraction with soba. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, pages pp. 2321–2324. ELRA, 2006.
[5] M. Babik M. Laclavik, M. Seleng. Ontea: Semi-automatic ontology based text annotation method. In *In Tools for Acquisition, Organisation and Presenting of Information and Knowledge*, ISBN 80-227-2468-8, Vydavatelstvo STU Bratislava, 2006.
[6] K. Bontcheva V. Tablan, D. Maynard and H. Cunningham. *Gate an application developers guide, http://gate.ac.uk/.*

[1] Department of Computer Science, Technical University of Cluj-Napoca, 15 Constantin Daicoviciu Street, 400020, Cluj-Napoca, Romania, tel. +4 0264 401 443
E-mail address: Ioan.Salomie@cs.utcluj.ro

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 123–126

EYE-TRACKING DATA EXPLORATION WITHIN INTERACTIVE GENETIC ALGORITHMS

MARCEL CREMENE[(1)], OVIDIU SABOU[(2)], DENIS PALLEZ[(3)], AND THIERRY BACCINO[(4)]

Abstract. This research is part of a project whose aim is to offer a general method for building artificial faces by using interactive genetic algorithm. Preliminary experiments are concerned with color discrimination and number comparison. The objective of this research is to compute the ranking for each individual (artificial face), based on the data collected from an eye-tracking system.

1. Introduction and problem statement

The idea proposed in this paper is related to a general method [2] for building artificial faces (police portraits), based on human interaction. An interactive genetic algorithm [1] creates artificial faces and displays them simultaneously on the screen. The idea is to create a ranking between these individuals and use a rank-based fitness assignment for the genetic algorithm.

An eye-tracking system records continuously the user's gaze activity (while looking at the virtual faces) in order to replace the human explicit actions like using keyboard or mouse. The eye-tracking system offers an interface that is much more faster and easier to use than a keyboard/mouse based interface and avoid the user fatigue (for a high number of iterations).

Preliminary experiments are concerned with color discrimination and number comparison. Thus, the rank is known a priori and we can use supervised learning in order to classify the ocular (eye-tracking) data.

Each subject involved in the experiments is asked to identify the lightest color (experiment 1) or the highest number (experiment 2). The eye-tracking systems measures various parameters such as: the time the user has focused on a colored square, the pupil diameter and its relative rank, the relative time focused on screen, the maximum variation of the pupil diameter, etc. In our experiments, there are 16 parameters used as inputs for the rank classifier.

Based on this measurements, the system assigns a (subjective) fitness to each candidate solution shown on the screen. Eight colors/numbers are presented to the screen simultaneously. Colors/numbers are ranked according to the subjective fitness. Each epoch, the fittest individual in the population is selected. Using this individual,

2000 *Mathematics Subject Classification.* 68T10 Pattern recognition, speech recognition.
Key words and phrases. Eye-tracking, Interactive Genetic Algorithm, Ranking, Classification.

the interactive genetic algorithm generates a new population. The process ends after a pre-established number of generations.

Our objective is to find a correlation between the ocular activity and the individual (color, number) rank. The aim is to train the system to discover the order the colors are presented in the experiment, based on the ocular data. Supervised learning technique is used. Data produced by the eye-tracking system is used as training data for a linear classifier in order to detect rules enabling to associate an unknown individual to a predefined class.

2. Proposed system

The system based on interactive genetic algorithms and eye-tracker interface is depicted in figure 1. The *Genetic algorithm* generates the individuals, the *Classifier* computes the rank for each individual, the *Individual Strategy selection* bloc uses the ranks in order to select the individuals that will be used for creating the next generations.

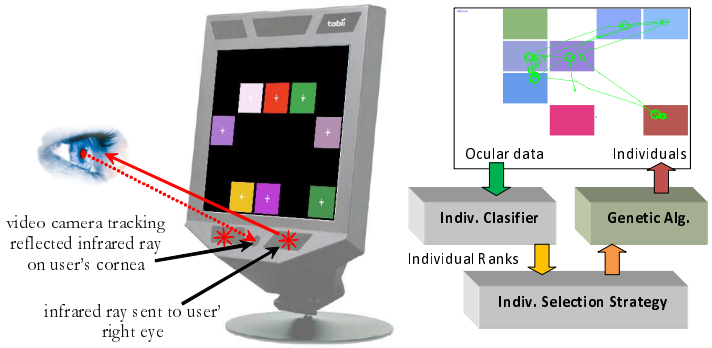


Figure 1. The system based on interactive genetic algorithms and eye-tracker user interface

3. Numerical experiments

A set of about 54000 data vectors was recorded from the eye tracking system. Each vector corresponds to one individual. We make the hypothesis that there is a correlation between the ocular activity and the individual rank. The rest of this section presents a set of tests that were done in order to test our hypothesis.

Test 1: MLP neural network classification. In order to classify the ocular data in 8 classes, a first test was done using a MLP neural network (Multi-Layer Perceptron) with 16 inputs and 24 respectively 14 neurons on the hidden layers, using the back-propagation algorithm. The test results with 5000 training vectors offered very poor results. The network was able to correctly classify only about 14% of the data (not included in the training set).

Test 2: multi-classifier toolbox. Classifier performance depends very much on the characteristics of the analyzed data. There is no classifier that works the best

on all possible problems. Knowing that, we have done also some additional data analysis using the toolbox "Matlab Classification Toolbox" from Meraka Institute.

We have analyzed our data using the following classification methods: Fisher's, Neural networks (NN), Naive Bayesian (NB), Gaussian (ML), Direct Tree (DT), and T-distribution with full covariance.

The classification errors obtained using the classification methods described previously are depicted in figure 2. A set of 10000 vectors was used. As we can observe, the best results were obtained using the Fishers linear classifier that it is also on of the fastest methods. On the second place we have the neural network classifier but the difference comparing to the Fisher's method is high. These errors were calculated for the training set of 10000 vectors.

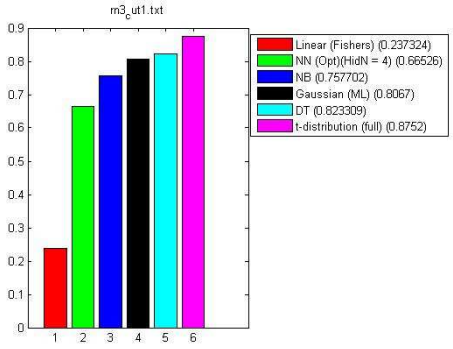


FIGURE 2. Data classification errors obtained with different algorithms

The toolbox includes some other algorithms such as k-Nearest-Neighbor and Gaussian Mixture Model but we were unable to use them on the full data set because they required too much resources (memory, time). However, we have tested these two algorithms on a reduced data set but their performances were poor.

Test 3: SVM classification. A multi-class implementation of SVM[4] was used to classify the dataset. The initial data was split in two parts, a training set of examples, comprising 5000 examples and the test set, containing the rest. The obtained accuracy was 32%.

After observing that the output of the classifier is not just a usual pattern, but a rank that forms an order relation with the rest of the classes, the accuracy analysis was further developed by taking into consideration the distance between the expected output and the one obtained from the SVM. An output result was considered accurate if it had an error less or equal than one. For example, if the expected output was 4 and the actual output was 3,4 or 5, the output was considered correct. For ranks close to the boundary (1, 2, 7, 8) the new accuracy was 81%. The middle ranks were classified with 55% accuracy. A possible explanation is that individuals of strong interest or no interest stand out from the set.

A binary version of the dataset, with the first three ranks in one class and the rest in the other class was also tested with SVM, but the results were actually worse

than the multi-class version. A possible explanation is that the SVM implementation used the one-versus-one strategy for multi-class classification which relied on multiple small machines with roughly equal number of training examples. The noise in the binary case contributed against the edge class, which had fewer training examples.

4. Conclusion and further work

The results that have been obtained so far are not accurate, but they indicate that individuals of interest could be separated from the rest by using ocular data. A possible improvement to the current approach is to reduce the dimension of the data (in order to reduce the noise) and classify the new data with SVM.

References

[1] Takagi, H., *Interactive Evolutionary Computation: Fusion of the Capacities of EC Optimization and Human Evaluation.* Proc. of the IEEE Vol. 89/9, 2001.

[2] Pallez, D., P. Collard, et al. (2007). *Eye-Tracking Evolutionary Algorithm to minimize user fatigue in IEC applied to Interactive One-Max problem.* GECCO (Genetic and Evolutionary Computation COnference), University College, London.

[3] G.J. McLachlan, *Discriminant Analysis and Statistical Pattern Recognition.* Wiley-Interscience; New Ed edition, August 4, 2004

[4] Chih-Chung Chang and Chih-Jen Lin, *LIBSVM : a library for support vector machines*, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

(1) Technical University of Cluj-Napoca, str. Baritiu, no. 26, phone: 0264-401813
E-mail address: cremene@com.utcluj.ro

(1) Babes Bolyai University of Cluj-Napoca
E-mail address: ovidiu@sabou.org

(3) University of Nice Sophia-Antipolis
E-mail address: denis.pallez@unice.fr

(4) University of Nice Sophia-Antipolis
E-mail address: thierry.baccino@unice.fr

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 127–130

DECOMPOSITION METHODS FOR LABEL PROPAGATION

LEHEL CSATÓ AND ZALÁN BODÓ

Abstract. In semi-supervised learning we exploit the "information" provided by an unlabelled data-set, in addition to the usually small training data-set. A commonly used semi-supervised method is label propagation [9] where labels are *propagated* from labelled to unlabelled data by employing similarity measures. The drawback of the algorithm is that its time requirement is prohibitive. This means that when a large amount of unlabelled data is used, a feasible algorithm is needed to compute the labels. In this paper we propose an approximation to label propagation. We divide the original problem into sub-problems that are computationally less prohibitive. A decomposition into K parallel sub-problems is considered where the sub-problems randomly and sparingly communicate with each other.

1. Introduction

Semi-supervised learning [9; 1] is a generalisation of the a pattern recognition problem to data sets where only a fragment of the available data is labelled. The motivation is that data labelling is a time consuming *human* activity whilst – in contrast – collecting unlabelled samples is cheap leading to huge amounts of unlabelled data, a good example is the data from DNA arrays [2] with only a tiny fraction processed, or the the huge document set from the internet, exploited by Google [5]. In semi-supervised learning the training data is augmented with unlabelled data, *i.e.* $\mathcal{L} \cup \mathcal{U} = \{(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_\ell, \mathbf{y}_\ell), \mathbf{x}_{\ell+1}, \ldots, \mathbf{x}_{\ell+\mathbf{u}}\}$, where ℓ and $\mathbf{u}$ are the sizes of the labelled and unlabelled parts respectively. We assume $\ell \ll \mathbf{u}$ and we use $\mathbf{n} = \ell + \mathbf{u}$. The task is to assign labels to the unlabelled part, using the *information* present in the joint data-set $\mathcal{L} \cup \mathcal{U}$.

Based on the *density* of the inputs, $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_\mathbf{n}\}$, and further restricting the *structure* of the data, we complete the labelling process. Suppose for example that "professor" is a good predictor for the *study* category. Then, if the words "professor" and "university" are correlated in $\mathcal{X}$, detection accuracy when using both words is improved. We use *label propagation* [9], a similarity-based technique where the labels are propagated based on closeness between data items. In this article we propose an approximation to handle large data-sets using ideas from stochastic sampling.

2000 *Mathematics Subject Classification.* 62H30, 68T05.
Key words and phrases. semi-supervised learning, kernel methods, label propagation.

2. Label propagation

Label propagation exploits the neighbourhood relation – the topology of the embedding space to construct a *graph* with nodes from $\mathcal{X}$ and edges encoding *similarities*; this graph *is used* to improve classification. Label propagation simulates a diffusion process that propagates the labels to neighbouring edges leading eventually to labels for the whole set $\mathcal{X}$. The graph is *fully connected* with edges weighted by the *degree of similarity* W_{ij} of the nodes $\mathbf{x}_i$ and $\mathbf{x}_j$:

$$W_{ij} = \exp\left(-\frac{d(\mathbf{x}_i, \mathbf{x}_j)}{\alpha^2}\right) \quad \text{and} \quad \boldsymbol{W} \stackrel{\text{def}}{=} \{W_{ij}\}_{i,j=1}^{n} \tag{1}$$

where $d(\cdot,\cdot)$ is a distance between the points and α is the radius of similarity. Other distance measures are the cosine similarity, Jaccard coefficient, Dice coefficient, or the similarity in (1) [3; 7]. In the following we use bold capitals for matrices and bold lowercase for vectors, other quantities are scalars. We normalise the similarity matrix $\boldsymbol{W}$, to obtain transition probabilities:

$$P_{ij} \stackrel{\text{def}}{=} \mathbf{D}^{-1/2}\, \boldsymbol{W}\, \mathbf{D}^{-1/2} \quad \text{where} \quad D_{ii} = \sum_{j=1}^{n} W_{ij} \tag{2}$$

with $\mathbf{D}$ diagonal. The resulting graph can be fully connected (see above) or sparse, these are trimmed from full graphs by cutting edges with small weights [8], here we use only full matrices. We define the following *label* matrices, assuming c classes (usually $c > 2$): $\mathbf{Y}_L$ an $(\ell \times c)$ matrix, each row corresponding to an item and each column to a category; $\mathbf{Y}_U$ a $(u \times c)$ matrix to be estimated; $\mathbf{Y} = [\mathbf{Y}_L^T,\ \mathbf{Y}_U^T]^T$ an $(n \times c)$ matrix – the concatenation of the above two matrices. Label propagation propagates labels using the following steps [9]:

(1) compute $\mathbf{Y}(t+1) = \mathbf{P}\, \mathbf{Y}(t)$.
(2) Reset the labelled data, $\mathbf{Y}_L(t+1) = \mathbf{Y}_L(0)$, set $t = t+1$ and go to (1).

When the iterations converge, labels for the unlabelled examples are given simply by taking the class with maximal label, an illustration is shown in Fig. 1. If multiple classes are desired, one can threshold $\mathbf{Y}_U$. It is interesting to note, that Google's efficient PageRank algorithm [5] works in the same way, except that label propagation is performed on a directed graph.

To analyse the algorithm we write the equilibrium solution as $\mathbf{Y}^* - \mathbf{P}\, \mathbf{Y}^* = \mathbf{0}$, where $\mathbf{Y}^*$ denotes the equilibrium solution and $\mathbf{0}$ is the vector of zeroes of length n. A subsequent step is to write the above algorithm as a constrained minimisation:

$$\begin{aligned} \mathbf{Y}_U^* &= \operatorname*{argmin}_{\mathbf{Y}_U} \mathbf{Y}^T (\mathbf{I}_n - \mathbf{P})\, \mathbf{Y} \\ &= \operatorname*{argmin}_{\mathbf{Y}_U} \begin{bmatrix} \mathbf{Y}_L \\ \mathbf{Y}_U \end{bmatrix}^T \begin{bmatrix} \mathbf{I}_L - \mathbf{P}_{LL} & -\mathbf{P}_{UL}^T \\ -\mathbf{P}_{UL} & \mathbf{I}_U - \mathbf{P}_{UU} \end{bmatrix} \begin{bmatrix} \mathbf{Y}_L \\ \mathbf{Y}_U \end{bmatrix} \end{aligned} \tag{3}$$

where the values $\mathbf{Y}_L$ do not change, with the exact solution [8]:

$$\mathbf{Y}_U^* = (\mathbf{I}_U - \mathbf{P}_{UU})^{-1}\, \mathbf{P}_{UL}\, \mathbf{Y}_L \tag{4}$$

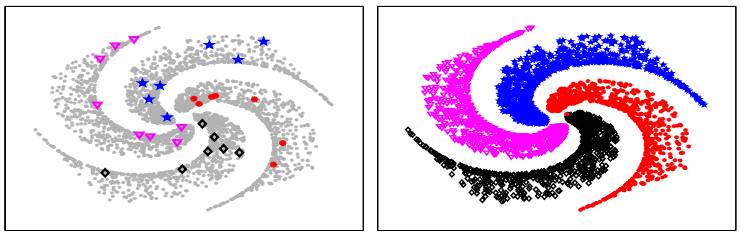


FIGURE 1. Illustration of label propagation: the left sub-figure shows the labelled data emphasised – there were 4 classes, and the right-hand side shows the resulting labels.

where we employed the matrix inversion lemma [4]. As we see, to have the solution, we have to compute the inverse of a large matrix, which has cubic computation time $O(u^3)$ and can be costly for the large data-bases that could potentially be exploited.

3. AN APPROXIMATE SOLUTION

We propose a decomposition for the above problem. For this first we consider the optimisation problem from eq. (4) and observe that the labels can be decomposed into c components, $\mathbf{Y} = [\mathbf{y}^{(1)}, \ldots, \mathbf{y}^{(c)}]$ and we have to solve the equations independently. It is therefore enough to focus on two-class case, that is on a single vector $\mathbf{y}^{(i)} \stackrel{\text{def}}{=} \mathbf{y}$. Let $\{A_1, A_2, \ldots, A_d\}$ be a *partition* of the unlabelled set. Let us decompose the *large* $\mathbf{L} \stackrel{\text{def}}{=} \mathbf{I}_n - \mathbf{P}$ in blocks. We denote with $\mathbf{L}_{k\ell}$ the block assigned to the pair A_k and A_l, *i.e.* $\mathbf{L}_{k\ell} = \{L_{ij} | i \in A_k; j \in A_\ell\}$ leading to:

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_L = \mathbf{y}_{A_0} \\ \cdots \\ \mathbf{y}_{A_d} \end{bmatrix} \qquad \mathbf{L} = \begin{bmatrix} \mathbf{L}_{00} & \mathbf{L}_{01} & \cdots & \mathbf{L}_{0d} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{L}_{d0} & \mathbf{L}_{d1} & \cdots & \mathbf{L}_{dd} \end{bmatrix}$$

and the minimisation from eq. (3) is written as:

$$\mathbf{y}^\top (\mathbf{I}_n - \mathbf{P}) \mathbf{y} = \sum_{a=0}^{d} \sum_{b=0}^{d} \mathbf{y}_a^\top \mathbf{L}_{ab} \mathbf{y}_b \tag{5}$$

where we stress that a and b start from 0 to include the labelled part of the data. We re-group the terms in eq. (5) to result in quadratic forms, each *within* a single partition A_k. Obviously, there is a *link* part that is responsible for the *global optimum*, the resulting expression – equivalent to eq. (5) – is:

$$\begin{aligned} \mathbf{y}^\top (\mathbf{I}_n - \mathbf{P}) \mathbf{y} \quad = \quad & \sum_{a=1}^{d} \begin{bmatrix} \mathbf{y}_0 \\ \mathbf{y}_a \end{bmatrix}^\top \begin{bmatrix} \mathbf{L}_{00} & \mathbf{L}_{0a} \\ \mathbf{L}_{a0} & \mathbf{L}_{aa} \end{bmatrix} \begin{bmatrix} \mathbf{y}_0 \\ \mathbf{y}_a \end{bmatrix} - (d-1) \mathbf{y}_0^\top \mathbf{L}_{00} \mathbf{y}_0 \\ & + 2 \sum_{a<b}^{d} \mathbf{y}_a^\top \mathbf{L}_{ab} \mathbf{y}_b \end{aligned} \tag{6}$$

The minimisation of eq. (3) is now equivalent with the minimisation of d *independent small local quadratics*, the label propagation problem taken on the labelled set and A_ℓ each and the minimisation of the *link terms* $\mathbf{y}_a^\top \mathbf{L}_{ab} \mathbf{y}_b$, this latter making the optimisation problem global. We note that the final eq. (6) is still *exactly* the original label propagation problem. We aim for solving the small problems separately and

then adjusting the original clusters according to the *fitness* of the clusters w.r.to the labelled set $\mathbf{y}_0$.

Algorithm. The proposed algorithm is stochastic minimisation that always finds the local optima within a single partition and updates the *partitions* such that the resulting subsets to be as uniform as possible. The algorithm is as follows:

- for $k = 1, \dots, d$ compute local optima $\mathbf{y}_k^*$;
- select pairs (k, ℓ) where we compute *pairwise* fitness of the solution, the last term in eq. (6): $-\mathbf{y}_k^\intercal \mathbf{L}_{k\ell} \mathbf{y}_\ell$;
- make adjustments if cluster solutions do not agree: swap data $\mathbf{x}_i$ and $\mathbf{x}_j$ that will lead to the largest increase in fitness, *i.e.* decrease in error.

References

[1] Y. Bengio, O. Delalleau, and N. Le Roux. Label propagation and quadratic criterion. In O. Chapelle, B. Schölkopf, and A. Zien, editors, *Semi-Supervised Learning*, pages 193–216. MIT, 2006.

[2] N. Cristianini and M. W. Hahn. *Introduction to Computational Genomics: A Case Studies Approach*. Cambridge University Press, 2006.

[3] U. v. Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[4] K. V. Mardia, J. T. Kent, and J. M. Bibby. *Multivariate Analysis*. Academic Press, 1979.

[5] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[6] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C*. Cambridge University Press, 1992.

[7] U. von Luxburg, M. Belkin, and O. Bousquet. Consistency of spectral clustering. *Annals of Statistics*, 36(2):555–586, 2007.

[8] X. Zhu. *Semi-supervised learning with graphs*. PhD thesis, School of Computer Science, Pittsburgh, PA, USA, 2005.

[9] X. Zhu and Z. Ghahramani. Learning from labeled and unlabeled data with label propagation. Technical Report CMU-CALD-02-107, Carnegie Mellon University, 2002.

Faculty of Mathematics and Computer Science, Babeş-Bolyai University Cluj–Napoca, Kogalniceanu 1, RO-400084

E-mail address: {lehel.csato, zalan.bodo}@cs.ubbcluj.ro

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 131–134

GROUP SELECTION IN EVOLUTIONARY ALGORITHMS

ALPAR PERINI[1]

Abstract. This paper proposes a new approach for selection in evolutionary algorithms. In this context, the fitness of a group is more important than the fitness of the individual. Each group evolves for a certain number of generations in isolation and only then is their fitness compared. The group fitness is obtained by averaging the fitness of a certain percentage of its individuals. This lesser number of evaluations is important, because the quality of such algorithms is compared after a fixed number of such fitness evaluations. This new algorithm was applied for obtaining the global minimum of functions defined in large dimensions.

1. Introduction

Evolutionary computation is a nature inspired optimization and search paradigm. It evolves a population of candidate solutions to a given problem. The selection of the fittest individuals in such algorithms is of utmost importance [1].

In our method, the fitness of a group is more important than the fitness of the individual. Each sub-population evolves for a certain number of generations in isolation, during which only a percentage of the members is evaluated. In the next step, a "competition" is made between the groups. Groups having better overall fitness can win members from the weaker ones, this way better mimicking nature and allowing for an intermixing between isolated groups.

It has been remarked that less promising individuals, manifesting altruistic behavior, can benefit evolution [4]. This novel model can promote the survival of such individuals, while the classical evolutionary algorithm clearly favors selfish behavior. The model was tested on some functions from the Large Scale Global Optimization (LSGO) Challenge [2].

Section 2 talks about some ideas that are somewhat related to our approach. Section 3 gives an overview of the model we have used for our method. Section 4 has some implementation details for applying our model to function optimization. Section 5 contains the experimental results that we have achieved. Section 6 presents some conclusions and future improvements.

2000 *Mathematics Subject Classification.* 68, 04.

Key words and phrases. group selection, evolutionary algorithm, fitness evaluations, selection method, global optimization.

2. Related Work

Group selection theory accounts for behaviors in societies that appear to benefit the group, even if it results in poor individual fitness [2]. A question arose that if the 'altruistic' individuals eventually die out due to low fitness, how can an altruistic trait continue to persist? Wilson argued [2] that it is possible for selective forces to favor traits that benefit the group, even if members are unrelated, as long as some group structuring conditions are met: the global population must be divided into temporarily isolated but periodically interacting subgroups with varying compositions.

In EAs, the population of individual solutions may be subdivided into multiple subpopulations [3]. Migration of individuals among the subpopulations causes evolution to occur in the population as a whole. In semi-isolated subpopulations, called demes, evolution progresses faster than in a single population of equal size. This inherent acceleration of evolution by demes could be confirmed for EAs and for genetic programming (GP) in particular [3]. Demes have common ground with our approach, nevertheless they do not refer to the way the selection is done and there is no competition for individuals within groups.

3. The Proposed Model

The new parameters of the model, compared to a classical genetic algorithm, are related to the number of groups, to the size of the groups, how many iterations to perform in isolation within a group, what percentage of the group should be used for evaluation and finally what percentage can a group loose from its members.

The group iterations can be executed in parallel for each group. They resemble a "mini" genetic algorithm with differences in selection and evaluation. During evaluation, only the given *percentage of members is evaluated* and these are the ones that contribute to the group fitness. As for selection, the individuals having no assigned fitness enter the competition with the *fitness of their group*.

After all "mini" GAs have exited, a *competition for individuals* is started between the resulted groups. Between two groups, the winner takes away at most a given number of individuals (percent of group size) from the looser group. This can help stronger groups evolve with better search space and penalizes groups with bad average fitness, possibly to the point of extinction.

This model can have the following benefits:

1. It reduces the number of fitness evaluations, which might be very costly for certain problems.
2. It is in favour of individuals manifesting altruistic behavior because at one step they might contribute to finding the optimal solution.
3. There is a periodical transfer of genetic material between isolated groups. This allows for better diversity and potentially to overcome local minima.
4. Isolated evolutions within groups can be done in parallel.

4. Implementation

We have implemented in Java our group selection model for finding the global optimum of large-dimensional functions [2]. All algorithm-specific parameters can be adjusted. The application lists some intermediate groups and the final population.

The chromosomes are encoded as real-valued arrays of dimension d: $x = (x_0, x_1, \ldots, x_{d-1})$. Fitness is assigned such that those that are closer to the minimum have better fitness values.

The initialization for the arrays is done with uniform random numbers. Within the "mini" GA, a q-tournament selection is used to populate the mating pool, two-point crossover is the recombination operator. Mutation range is adapted using the Normal distribution.

We have chosen to keep the known best individual from the current generation in the new one, not to loose a known potential good solution.

The evaluation of a group involves a percentage of its members. Individuals are chosen uniformly random and they are evaluated by making a call to the test function, which costs one fitness evaluation (FE). In case a call is made when no more function evaluations remain, an exception is propagated. The group fitness is the average fitness of the individuals chosen for evaluation.

The competition between resulting groups is done with binary tournament regarding group fitness.

Finally, the algorithm terminates when no more fitness evaluations are available. At this step, all groups are unified, sorted decreasing by fitness and printed.

5. Experimental Results

Testing was carried out on the large-dimensional functions given at the 2008 LSGO competition [2]. Here a fixed number of FEs are allowed for each problem: $5000 \times Dimension$.

We show the test results for functions F1-F6 in 1000 dimensions. We have worked with 10 groups having 100 individuals each. The tournament size varied from 5 to 7. 10 group iterations were done in isolation. 10% of each group was used for group evaluation. A group can loose up to 10% of its individuals at one iteration.

Table 1 below illustrates the results we obtained for each test function. Based on the constraint on the number of evaluations, 5 million FES were permitted. OptMin is the optimal minimum of the function that is known. Error represents the difference between the optimal and the obtained minimum. StdDev is the standard deviation of the results that were obtained. ErrMin and ErrMax refer to the best and worst error result obtained at the LSGO competition.

The accuracy of the results varied very much depending on the function that was tested. F5, Shifted Griewand was optimzed the best, while F3, Shifted Rosenbrock was optimzed the worst.

Func	*OptMin*	*Error*	*StdDev*	*ErrMin*	*ErrMax*
F1	-450	1.89E+00	3.66E-02	0.00E+00	3.58E+04
F2	-423	8.00E+01	7.20E-01	1.04E-05	1.47E+02
F3	390	34.21E+02	3.39E+02	3.41E-04	8.98E+09
F4	-330	2.73E+00	3.83E-02	0.00E+00	1.03E+04
F5	-180	8.66E-03	6.72E-07	0.00E+00	3.04E+02
F6	-140	9.11E-02	2.76E-05	4.26E-13	1.99E+01

TABLE 1. Results obtained for 1000 dimensional functions F1-F6 from LSGO .

6. Conclusions and Future Work

This paper presented a novel approach for improving the existing genetic algorithms. The main purpose was to reduce the number of FEs while avoiding to get stuck at local minima. Secondly, the proposed selection method allows for altruistic individuals in a natural way.

Experimental results for very difficult test functions show this is a promising model but, as with any new approach, requires a lot of fine-tuning for all its many parameters. A key aspect is, that even though the fitness is only known for a fraction of the overall individuals, the implemented algorithm has shown signs of good convergence.

Finally, introducing more sophisticated genetic operators can also have an impact on performance.

References

[1] D. Dumitrescu, B. Lazzerini, L.C. Jain, A. Dumitrescu. *Evolutionary Computation*.
[2] Ke Tang. 2008. *Summary of Results on CEC08 Competition on Large Scale Global Optimization*. University of Science and Technology of China
[3] Markus Brameier, 2004. *On Linear Genetic Programming*. Dortmund University.
[4] Momoko Price. 2006. *The Controversy of Group Selection Theory*. The Science Creative Quarterly, Jan-Mar 2006.

[1] Department of Computer Science, "Babes-Bolyai" University of Cluj-Napoca, Romania
E-mail address: pai11272@cs.ubbcluj.ro

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 135–138

SUSTAINABLE DEVELOPMENT GAME

ANDREI SÎRGHI[1]

Abstract. A game-theory approach to the sustainable development problem, using a game with three players: *economy*, *environment* and *society*, is proposed. Standard sustainable development methods are based on models which analyze the sustainable development of separate areas of a region (e.g. country or city), and then tries to integrate the results to achieve a complete vision. Proposed approach introduces a new mathematical model which represents the decision process of all areas in a single system that is modelled using a game. A key element of this approach is the impact of each game player on the other players involved in sustainable development game, which is analyzed using Game Theory. This makes approach very appropriate to the real life processes, where decisions for every area are made in conditions of conflict and are based on the behaviour of the competitors.

Introduction

The concept and methodology of sustainable development (SD) [DalalBass] appeared over the past few decades as a result to a set of interdependent issues like: climate change, pollution control, preservation of biodiversity and water resource management.

The paper describes a mathematical model for the decision process within SD problem. This model is called sustainable development game (SDG) and it is based on Game Theory principles. Proposed model is intended to show how much the decisions taken in one area influences the other areas of a region. SDG model represents the decision process within SD by a three player game. The goals of players are contradictory, and the game equilibrium is achieved when players find a compromise solution.

1. Sustainable Development Problem

The concept of sustainable development was used in 1987 by *Brundtland Commission* in [WCED], which formulated what became the most used definition of SD as *"development which meets the needs of the present without compromising the ability*

2000 *Mathematics Subject Classification.* 91Axx, 91Bxx.
Key words and phrases. Sustainable Development, Game Theory, Complex Decisions.

of future generations to meet their own needs". At the base of the concept is the principle that objectives of society, economy and environment should be complementary and interdependent in the development process of a region.

The problem related to SD which we propose to solve in this paper can be defined as:
Create a mathematical model having the next characteristics:

(1) Represents the decision model for a real development process of a region from the three aspects corresponding to major areas: economy, environment, society;
(2) Valuable in the real decision making process;
(3) Every area has a particular set of objectives and decision functions;
(4) A solution model represents a compromise solution between all areas.

Our aim is to develop such a model by considering a specific region and supposing to have complete information about economy, environment and society.

2. Related Work

SD problem received high attention from its first apparition, but until now does not exist a powerful mathematical model that can be used to represent this problem. The result of researches of almost all communities that analyze this problem is a huge set of indicators that can be aggregated to give a mark for the quality of countries sustainable development process.

There are two widely accepted methods to measure the sustainable development of a country:

1. Sustainable Development Gauging Matrix (SDGM) [Zgurovsky]. The measure technique of SDGM consists in the aggregation of three dimension indices: economic (I_{ec}), ecological (I_e) and social (I_s) in the index of sustainable development (I_{sd}). Further, each of these indices is calculated by using other six global indices widely used in Statistics communities.

2. IPAT equation [Chertow]. Expresses the relationship between technological innovation and environmental impact. $IPAT$ states that human impact (I) on the environment equals the product of population (P), affluence (A: consumption per capita) and technology (T: environmental impact per unit of consumption).

3. Interaction Model

The SDG model can be built as a system of models for sustainable development areas: economy, environment and society. The dependencies between particular area models, are very valuable for our approach. They result in a huge impact to SD. In this section we give an abstract system of dependences between economy, environment and society.

The economy plans the optimal amount of products outcome by choosing corresponding quantities of natural resources ($E(t)$) and social capital ($S(t)$).

The function of economic development (denoted EC) may be represented as a dynamical system given by a particular *production function* [Mishra] *ec*:

$$EC(t+1) = ec(E(t), S(t)), \tag{1}$$

where t stands for time period, $t=\overline{[t_0,\ T]}$.
The sustainable activity of economy can be measured with the *economical sustainability index* (I_{ec}) from SDGM.

Environment tries to clear polluted renewable resources and to achieve a sustainable trajectory in environment development. Environment development may be represented as a dynamical system which represents the evolution of natural resource stocks:

$$E(t+1) = e(R_n(t), R_r(t)), \tag{2}$$

where $R_n(t)$ and $R_r(t)$ are stocks of nonrenewable and renewable resources in period t.
The development of environment is measured with *environmental sustainability index* (I_e), designed by the Center of Ecological Legislation and Policy of Yale University (USA), which is also included in SDGM.

The environment and the economy influence the living conditions in region, which can be suitable or not for people life. Analyzing these conditions, the society has to choose, to stay in this system or not.
The function of social development may be represented as a dynamical system:

$$S(t+1) = s(E(t), EC(t)). \tag{3}$$

The society development can be measured with *social sustainability index* (I_s) included in SDGM.

4. Sustainable Development Game

Within SDG model, Game Theory [OsborneRubinstein] is used to represent the decision process of SD problem. The SD game involves three players: the economy (EC), the environment (E) and the society (S).
Player EC has two strategies:

(1) to choose a quantity of resources that follow environmental standards (ES), or
(2) to use *exhaustingly* environment resources (NES).

Second player in the game represents the environment, and has two strategies:

(1) to be suitable for human life and for economy (ST) (to restrict as much as possible resource consumption), or
(2) not suitable (NST).

The third player is society, with strategies:

(1) to stay in this system, i.e. to live and to work here (L), or
(2) to evade from it (NL).

In SD game, economy decides the first move by choosing either an economic strategy that follow environment standards (ES), or one strategy that destroys the environment (NES).

The environment move can be suitable for people life and for economy (ST) or not (NST). But environment is not informed about EC choice, it has just a belief about behavior of EC. Selecting an appropriate strategy, the environment imposes the admissible values of resource usage for EC.

Eventually, the society must choose to live in this system (L), or not (NL), without any information about the other players moves excluding its belief.

The main idea behind the game is that every agent must *"move carefully"*. If an agent will follow a strategy that influences negatively other agents, then the opponents will limit the future actions of the agent by imposing direct and indirect penalties.

Conclusion and Further Work

Game Theory is a suitable tool to analyze the behavior of players involved in Sustainable Development and to find compromise solutions between individual plans. It makes proposed model different from others because it can be used to analyze, predict and control the development of a region in conformity with sustainable criteria, when other models just propose methods to evaluate the state of sustainability of a region.

Further, we want to integrate SDG approach with different models for: environment, economy and society development, to analyze the interactions between these areas in different situations.

References

[Chertow] Chertow M. R., "The IPAT Equation and Its Variants", *Journal of Industrial Technology*, Volume 4, Number 4, pp. 13-29, MIT Press, Cambridge, 2001.

[DalalBass] Dalal-Clayton B., Bass S., "Sustainable Development Strategies", Earthscan Publications Ltd, London, 2002.

[Mishra] Mishra S.K., "A Brief History of Production Functions", Social Science Research Network (SSRN), Shillong, 2007.

[OsborneRubinstein] Osborne M. J., Rubinstein A., "A course in game theory", MIT Press, Cambridge, 1994.

[Zgurovsky] Zgurovsky M. Z., "Sustainable development global simulation: Opportunities and treats to the planet", National Technical University of Ukraine "KPI", Kyiv, 2007.

[WCED] WCED, "Our common future", *The Brundtland Commission Report, World Commission on Environment and Development*, Oxford University Press, Oxford, 1987.

[1] Cluj-Napoca, Romania
E-mail address: andreisirghi@yahoo.com

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 139–142

DESIGNING SEARCH STRATEGIES FOR ROBOTS USING GENETIC PROGRAMMING AND MICROSOFT ROBOTIC STUDIO

CLAUDIU SORIN IRIMIAŞ

Abstract. Abstract-This paper presents a method for automatically generating computer programs to search the position of a particular object into a room. The goal of this research is to prove that genetic programming (GP) can be used together with Microsoft Robotic Studio (MSRS) runtime to evolve a hierarchical control architecture for a simple navigation on a autonomous robot. The results summarized in this paper present a control architecture evolved for a Lego Nxt Tribot robot simulated in Microsoft Visual Simulation Environment (MSVSE).

1. Introduction

This paper presents a method for automatically generating computer programs able to perform the task of establishing the position of a particular object into a room. At this moment mobile robots are usually programmed manually by a programmer. This approach introduces many difficulties due to the complexity of the task in a real word environment [2]. Instead, GP will considerably reduce the problem, to the point to find an appropriate fitness function that can describe how well a particular individual solves a task.

The use of a GP method to evolve controller architectures for robots has been reported previously in various ways. GP was used to program a robot to move a box located in the middle of an irregular shaped room to the nearest wall [1]. Another variant is controlling GP system to evolve obstacle avoiding behavior in a sense-think-act context [4, 5]. An alternative to standard GP, that applies layered learning techniques to decompose a problem, was used to evolve agents to play keepaway soccer, a sub problem of robotic soccer that requires cooperation among multiple agents in a dynamic environment [6].

2. Objective

The objective purposed to be achieved by the robot is to locate an object (i.e. a box) in a room only using two bumpers and a web camera. One of the bumpers

2000 *Mathematics Subject Classification.* 68T, 70E.
Key words and phrases. search strategies, robotics, genetic programming, microsoft robotic studio.

is planned to be attached in the front of the robot and the other on the rear part. The web camera is not used to locate the box from the beginning, because image interpretation requires more processing. And, in this case, the web camera can be easily replaced by another sensor that is capable to identify an object (like a sonar sensor). The searched item will have a color that could be recognized by the web camera sensor and no other obstacle, not even the rooms wall has the same color.

3. Representation

The chosen representation was proposed to be as simple as it could, so that it might be used with a small population size but with a reasonable number of generations and possible moves.

Terminals are used to provide the movement control of the robot. The terminal set is formed from the primitive operations: MoveForward (T1), MoveBackward (T2), MoveLeft (T3), MoveRight (T4), MoveForwardLeft (T5), MoveForwardRight (T6), MoveBackwardLeft (T7), MoveBackwardRight T(8). The following terminals are used to verify the state of the environment and of the autonomous robot: FrontBumperPressed (T9), RearBumperPressed(T11) and BoxHit (T12).

The **primitive function** used to connect the primitive terminals is a conditional primitive function. The structure of this function is: *Condition(StateTerminal, MovementTerminal, MovementTerminal)*, where StateTerminal –is one of the terminals T9 –T10, MovementTerminal –is one of the terminals T1 –T8. The conditional function will evaluate the StateTerminal and depending on result obtained it executes the first MovementTerminal (StateTerminal evaluated to true) or the second MovementTerminal (StateTerminal evaluated to false).

The **fitness function** tries to minimize the number of moves needed to locate the object in the room. If the robot finds the searched object the corresponding individual will be rewarded with a big value. The chromosome will be penalized with the number of moves made and with the number of times the robot crosses the same point. The surface of the room is seen as a two dimensional matrix. The starting point of the robot is considered to be on line and column 0. Depending on the move the robot makes, the position in the matrix will be updated. At one point the line and column values could be increased or decreased by one unit.

After the robot finishes all his movements, applying the rule from above, will be obtained a matrix having as elements the coordinates visited by it. Using this information we can penalize the program generated if the robot has been at the same location more than once. Tests have shown that the value, with which the individual should be penalized, in case the robot passes through the same location more than once, should be the number of passes through the same point multiplied by 100. So for computing the fitness of one individual was used the following formulae: $F = R - MP - CP$, where F –the fitness value, R –the reward, which will be equal with 10000 if the object, was found or 0 otherwise, MP –moves penalization, CP –cross through the same point penalization.

4. Framework

To evaluate how well an individual from the population is performing the moves generated for him must be executed by the robot. After that, using the state of the robot and of the environment the fitness function could be computed. The communication between the GP algorithm and the robot is done by using a framework that recognizes the movement types required by the individual.

The functionality of the framework is straightforward: 1) The genetic program has to evaluate the terminals for an individual. 2) These terminals are transformed into functions known by the framework. The transformation is done by using a module common between the framework and the genetic program. 3) The functions obtained are written into a file from the framework, using a predefined file template. 4) The new changes are compiled. 5) The simulation is launched and the new moves are sent by the framework to be executed by the robot, one by one. 6) After each execution the environment and the robot state are updated. 7) The final state of the environment and of the robot is sent to the genetic program. 8) The genetic program computes the fitness function of the individual based on the information received from the framework.

5. Experimental Results

This section presents some of the results obtained in different scenarios and will focus on two specific experiments. The algorithms presented here were implemented in C# programming language and run on AMD Athlon 64 Processor 1.8 Ghz, 1.5 GB of RAM machine running in Windows Xp environment. 10 experiments were performed. The experimental runs were made on 25 generations with a population size of 10. Roulette wheel selection was used and the crossover, reproduction and mutation probabilities were set to 90%, 10% and 10% respectively. The maximal number of terminals in a chromosome was set to 100. The simulated room was made to have a square form having the entire area of 10 m2. The robot was placed in the lower part of the simulated room, relatively close with the wall, faced with the center of the area. The searched object is a red cube having one side equal with 1 m. In each move the robot advances or retreats with 0.25 m powering up the motors at 50%. When a robot makes a turn the actual power for the motors is set to be at 7%, so that the move can be more accurate.

From the 10 experiments run, only 4 evolved into a solution. In all of the solutions the objective was achieved using just a part from the set of moves generated for that individual. However, even if a solution was generated in 4 cases, none of them was the optimal solution. From the resulted data I have noticed that the individual from the population leans to the best individual starting, on average, with the 20th generation.

In what follows I will present in more details test number 3, from the 10 tests presented before, because it was the first to obtain a solution for the problem. In the solution the box was found after 20 moves: *ForwardLeft, Left, Backward, Backward, BackwardRight, Forward, ForwardLeft, Forward, Forward, ForwardRight, Forward, Forward, ForwardLeft, ForwardLeft, Forward, Forward, Right, Forward, ForwardLeft, Right*. The best individual from the 3rd test evolved into a solution, generation by

generation. It can be seen how the best individual of the population had the fitness equal to -214 until the 7th generation. Starting with the 8th generation the population fitness has begun to increase up to -86. Then it evolved to -9 in generation 9. And evolves once again to 8250 in generation 21, remaining the best population fitness until the end of the generations.

6. Conclusions and future work

This paper has shown that GP can be used to evolve a hierarchical control architecture for a simple navigation on a autonomous robot using the MSRS runtime. The results have shown that the optimal solution was not found, which means that GP algorithm will need more evolutionary time to come up with an improved result. The next step will be to experiment the proposed system on a real environment with a real Lego Nxt Tribot robot. Also, based on the experimental results described above, I am currently enhancing the implementation for a new objective for the robot, the task of moving a box from the middle of an irregular shaped room to the wall [3].

References

[1] J. R. Koza, Genetic Programming, MIT Press, Cambridge, Massachusetts, 1992.
[2] R. A. Brooks, A robust layered control system for a mobile robot, IEE Journal of Robotics and Automation, RA-2(1):14-23, March 1986.
[3] J. R. Koza, J. P. Rice, Automatic Programming of Robots using Genetic Programming, MIT Press, Cambridge, Massachusetts, 1992.
[4] E.V. Siegel, J.R. Koza, Genetic Programming Controlling a Miniature Robot, MIT, Cambridge, Massachusetts, 1995
[5] M.C. Martin, Genetic Programming for Robot Vision MIT, Cambridge, Massachusetts, 2002
[6] S.M. Gustafson, W.H Hsu, Layered Learning in Genetic Programming for a Cooperative Robot Soccer Problem, Springer-Verlag, 291-301, 2001

Department of Mathematics andComputer Science, Babeş-Bolyai University Cluj-Napoca, Kogalniceanu 1, 400084 Cluj-Napoca, Romania
E-mail address: sorin_irimias@yahoo.com

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 143–146

MODELING MULTIAGENT IRATIONAL ALGORITHMS FOR GAMES

OVIDIU ŞERBAN

Abstract. Sometimes, theoretical optimal strategies are not the best choice because in nature, a perfect solution of a problem can be found only when several parameters of the given phenomenom are ignored, and game-theory is no exception. By using multiagent systems, it is possible to solve a number of problems that are unsolvable using traditional game-theory. In this paper I will try to present that irrational algorithms are better than rational ones, not due to their performance, but due to the fact that they easily simulate a human behaviour. The game played by the agents is an auction game, that could be transformed into a real business game.

1. Introduction

Games have long been a popular area for research in AI, because games are challenging yet easy to formalize, they can be used as platforms for the development of new AI methods and for measuring how well they work. In addition, games can demonstrate that machines are capable of behavior generally thought to require intelligence without putting human lives or property at risk. Also, in the past years some research fields were interested in simulating emotions of computer software [1].

The so-called good old-fashioned artificial intelligence techniques [2] work well with symbolic games, and to a large extent, GOFAI (Good Old- Fashioned Artificial Intelligence) techniques were developed for them. GOFAI techniques have led to remarkable successes, such as Chinook, a checkers program that became the world champion in 1994 [4], and Deep Blue, the chess program that defeated the world champion in 1997 and drew significant attention to AI in general [3].

Also, machine-learning techniques, such as neural networks, evolutionary computing and reinforcement learning, are very well suited to video games. Machine Learning techniques excel in exactly the kinds of fast, noisy, numerical, statistical, and changing domains that today's video games provide. These kind of techniques can be easily modeled using multi-agent systems, or in other cases they can be described using a mathematical aproach. Our approach in this paper is well suited

2000 *Mathematics Subject Classification.* 91A30, 91A60.
Key words and phrases. Game Algorithms, Realtime games, Agents, Inteligent Agents, Artificial Inteligence, Gaming Strategies.

to be expanded for economical games, where players are afraid to "play" against a computer software and when they discover one, it could mean the end of it.

2. The Game

The game can be described by making a parallel with an auction. The target of the game is to maximize the portfolio value in a given budget. The value of an object and its initial price is known from the beginning, so every player can make its own strategy to win the game.

We will consider to be the best strategy, the optimal one generated by a single player game, so that the last price of an object is equal to its initial price. Also we suppose that the number of optimal strategies is less than the number of players partincipating in an auction, so that is encouraged the competition between them, and no one can choose the optimal strategy. In most cases the number of optimal strategies is equal to one. The objects for the auction are uniformly distributed over the game, meaning that utility ratio(given by *value/initial price*) is distributed uniformly over the collection of objects and so a greedy strategy (where the decision is given by utility ration) won't win. Despite this, the greedy strategy is used for comparison with the obtained strategies. Each round a certain player begin the auction, and the round ends when all the objects were auctioned. To make all players' chances equal, the starter is chosen randomly every round.

3. Algorithms

Algorithms are modeled as a collection of decisional parameters, so the irrationality of the model is given by the possibility of hesitation or withdrawl if the bet is to high. The reason of modeling an irrational algorithm is that a human player don't want an computer oponent to be perfect and he can be sometimes afraid of such an algorithm. So we are trying to model emotion in a constraint based manner.

One of the possibility of is to have three kind of parameters, that should model fear, risk and the ability of prediction. Fear is responsible for the hesitation of taking a decision and modeled in a matematical manner, the fear apears when the remaining buget is low or there are few remaing objects so you should raise the bet. Risk is a difficult to model factor, but if you are taking the term in a more general manner, the risk is modeled as the pushing your buget to the limits by raising a price, or by obtaing the most valuable object with any price. Prediction is the ability of predicting a good bet for a certain object, based on its value.

The algorithms are obtained in an evolutionary[5] manner by beginning with a certain population and grouping players into a group. From that group, after running a game, you should obtain the two best players, named masters, which would create another player, as a combination of the two masters. Also the resulting player can

support a mutation of its characteristics. The two masters and the result player are added to the new population. Decision can be modeled using the following formulas:

$$remainingBuget/initialBuget < riskPx \mapsto raiseFactor \tag{1}$$

$$(currentBet - objectsPrice)* \\ (1 - predicitionPx) \mapsto predictionPrice \tag{2}$$

$$increase \leftarrow currentBet + predictionPrice - objectsPrice \\ increase/objectsPrice > fearPx \mapsto fearFactor \tag{3}$$

$$decrease \leftarrow remainigBuget - (currentBet + predictionPrice) \\ decrease/remainingBuget < fearPx \mapsto fearFactor \tag{4}$$

where riskPx (1), predictionPx (2) and fearPx (3)(4) are chosen at the initialisation of the algorithm. The three factors determine an decision that can be also probabilistic.

4. Conclusions

Since the 1990s, the field of gaming has changed tremendously. From modest sales in the 1960s [6], sales of entertainment software reached $25.4 billion worldwide in 2004 [7]. Curiously, very little AI research has been involved in this expansion. Many video games do not use AI techniques, and those that do are usually based on relatively standard, labor-intensive scripting and authoring methods.

The techniques described in this paper are probable not so realistic and they represent the begining of an emotional model, dedicated to produce algorithms capable to play in a more human manner. Human players desconsider in almost all situations the "perfect players", so if the AI simulates fear and maybe hesitation, the algorithm will be more challenging than a perfect one.

References

[1] Research problems in the use of a shallow AI model of personality and emotion, http://condor.depaul.edu/ elliott/papers/aaai94n/aaai94n.html, last visited on 20.01.2009
[2] Haugeland, J; Artificial Intelligence: The Very Idea; Cambridge, Mass.: MIT Press, 1985
[3] Campbell, M., A.J. Hoane Jr., and F. H. Hsu; Deep Blue. Artificial Intelligence, 2002
[4] Schaeffer, J; One Jump Ahead; Berlin: Springer-Verlag, 1997
[5] Goldberg, David E., Genetic and evolutionary algorithms come of age, Journal Commun. ACM, 3/37, pages 113-119, ACM, 1994
[6] Baer, R.H; Videogames: In the Beginning; Springfield, N.J.: Rolenta Press, 2005
[7] Crandall, R.W., and J.G. Sidak; Video Games: Serious Business for Americas Economy; Entertainment Software Association Report, 2006

Faculty of Mathematics and Computer Science, Babes-Bolyai University, Kogalniceanu 1, Cluj-Napoca, 400084, Romania.

E-mail address: soi11275@scs.ubbcluj.ro

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 147–150

COEVOLUTION FOR FINDING SUBGAME PERFECT EQUILIBRIA IN 2-PERIOD CUMULATED GAMES

RADU M. BERCIU[(1)]

Abstract. Based on strategic games a new type of dynamic game is introduced, n-Period Cumulated Game, where players engage in a repetitive play of a constituent strategic game for n number of times (an accumulation period), without receiving their payoff after each stage of the game, but only the cumulated payoffs of all the stages of the game at the end of each accumulation period. Then we apply a two-population based Genetic Algorithm in order to find Subgame Perfect Equilibria in 2-Period Cumulated Games.

1. Introduction

The idea to bring together Genetic Algorithms (GAs), a well known optimization method introduced by John Holland in the early 1970s [1], and Nash strategy [2][3], the most commonly encountered solution concept in Game Theory, in order to make the genetic algorithm build the Nash Equilibrium belongs to Sefrioui [4]. As described in [7] at each generation a player improves its strategy with respect to the other players' best strategies of the previous generation: Nash Equilibrium is reached when no player can improve its strategy. Based on this, NCA (Nash Coevolution Algorithm), a two population based GA, finds subgame perfect equilibria in 2-Period Cumulated Games interpreted as extensive games with imperfect information.

2. n-Period Cumulated Game (n-PCG)

The notion of cumulated game basically means that, for instance, having two individuals playing a number of strategic games, there exists a mechanism that sums and withholds the benefits until the players have completed n plays of the strategic game; n is called the length of the accumulation period. Their accumulated benefits are reported only after n stage games. Our goal is to model the situations where players engage in games in which they have different knowledge of the previous plays, and in which they receive their payoff after different accumulation periods. For example, consider the model of the relation between an employer and an employee where the employer agrees to pay the employee a small amount of money every two weeks even

2000 *Mathematics Subject Classification.* 68T20, 91A25, 91A20, 91A18, 91A10, 91A05 .
Key words and phrases. n-Period Cumulated Game, Subgame Perfect Equilibrium, Coevolution, Game Theory.

though he may gain benefits from the employee's work at every two months. The accumulation period of the employee is two weeks and the accumulation period of the employer is two months. However, here the focus is on the notion of n-Period Cumulated Games (i.e. equal accumulation periods), and on the analysis of four models of 2-PCGs distinct by their elementary game.

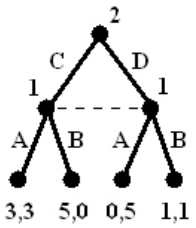
The notion of cumulative benefit game appeared before in [9]. There the author argued that in the context of repetitive games and a strong temporal discounting, accumulation can promote a cooperative strategy. However an n-PCG is a not a repetitive game even though it is a dynamic one.

2.1. **n-Period Cumulated Games as dynamic games.** An image of the described game is that of a dynamic game. A dynamic game captures the idea that players act sequentially and can incorporate previous information about earlier moves in the game in choosing their next move. Even though a stage game (called constituent game) is being repeatedly played, the difference between an n-Period Cumulated Game and a finitely repeated game is that the benefit/payoff is reported only after an accumulation period.

The n-PCG can be represented as an extensive game. For instance the static game in FIGURE 1(A) is equivalent to the dynamic game represented in FIGURE 1(B). The dashed line between some nodes means that the current decision maker does not know in which state she is at. Therefore the extensive game can be viewed as a 2×2 strategic game where players act simultaneously. This is the case of extensive games with imperfect information, i.e. each player, when making a decision, is not perfectly informed about the events that have previously occurred [5][8].

	C	D
A	3,3	0,5
B	5,0	1,1

(a) the game in its normal form



(b) the game in its extensive form

FIGURE 1. Views of the same 2×2 Strategic Game

3. Solution Concept for n-Period Cumulated Games

Interpreting the n-Period Cumulated Game as an extensive game the adequate solution concept is that of Subgame Perfect Equilibrium. A subgame is a subtree from a game's directed tree: that begins at a decision node, that gives to the initial player all the decisions that have been made until that time, and that contains all the decision nodes that follow the initial node. The Subgame Perfect Equilibrium (or Subgame Perfect Nash Equilibrium) is a refinement of Nash Equilibrium(NE)

that induces a NE in every subgame (subtree) of that game. In an n-PCG there are subgames with simultaneous decisions, therefore all possible Nash Equilibria of that subgames may appear in a Subgame Perfect Nash Equilibrium.

4. NCA (Nash Coevolution Algorithm)

A coevolution algorithm for finding Subgame Perfect Equilibria in 2-Period Cumulated Games is proposed. This algorithm is called Nash Coevolution Algorithm (NCA).

4.1. **Encoding a strategy.** With NCA a chromosome represents a strategy that is a function of the state of the game: every possible state(history) has one slot i,$0 \leq i < s$, in the s^m string that codes the move that the player will take if she uses that strategy and she encounters the history with index i;s is the number of states of the constituent strategic game (stage game), m is the length of the history (the number of stage games that the player recalls) and the hypothetical game is there to induce the first actions that the player will take. The difference from [6] is that here there is no restriction for m.

4.2. **Fitness Assignment.** Every player is represented by a population that at every generation maximizes the player's payoff. The populations evolve by optimizing at each step their chromosomes using the other population's best q chromosomes from the previous generation [7], i.e. each player has a population that tries to maximize its payoff using the best solutions found by the other player's population one generation before. The fitness of each chromosome will be the mean payoff it receives after playing with each of the other player's best chromosomes from the previous generation.

5. Convergence, stability and more

Different parameters configuration gave an insight into the stability of NCA and proved empirically that

$$stability \approx \frac{1}{exchange \times p.m.},$$

where *exchange* is the number of individuals used to evaluate a strategy and $p.m.$ is the probability of mutation.

The convergence time to an equilibrium is between 40 and 60 generations for all the configurations for the Leader game. For the Battle of sexes, for the configurations 1 to 9 in more than 90% of the cases an equilibrium was found in less than 70 generations; for the last three test configurations the convergence time to an equilibrium is under 40 generations. For the Prisoner's Dilemma game it needs less than 30 generations to achieve its equilibrium in all the configurations. For the Hawk-Dove game the algorithm achieves an equilibrium under 70 generations with a probability of 87%.

6. Conclusion and Future Work

Based on strategic and repetitive games a new dynamic game called n-Period Cumulated Game has been introduced. Tests on simple forms of these games showed that in a coevolutionary environment (provided by the developed NCA) players eventually play the Nash Equilibrium in every subgame, even though they are not aware of their payoff until a certain number of stage games are played.

An interesting focus for future work is that of working with different levels of accumulation periods, testing the outcome of having the players engage in these basically different games. Another challenging issue can be that of letting two players play a number of games between each other, thus transforming the n-PCG into a repetitive one; for instance, preliminary tests show that in the repetitive n-PCG based on the Prisoner's Dilemma the players start by cooperating.

References

[1] Holland J. H. Adaptation in natural and artificial systems. *MIT Press,Cambridge, MA, USA*, 1992.
[2] Nash J.F. Equilibrium points in n-person games. *Proc. Nat. Acad. Sci.*, 36:48-49, 1950.
[3] Nash J.F. Noncooperative games. *Annals of Mathematics*, 54(2), 1951.
[4] Sefrioui M. *Algorithmes evolutionnaires pour le calcul scientifique: Application a l'electromagnetisme et a la mecanique des fluides*. PhD thesis, Universite Pierre et Marie Curie, Paris, 1998.
[5] Rubinstein A. Osborne M.J. *A Course in Game Theory*. MIT Press, Cambridge, MA, USA, 1994.
[6] Axelrod R. The evolution of strategies in the iterated prisoner's dilemma. *Genetic Algorithms and Simulated Annealing*, pages 32–41, 1987.
[7] Periaux J. Sefrioui M. Nash genetic algorithms: examples and applications. *Proceedings of the 2000 Congress on Evolutionary Computation*, pages 509–516, 1999.
[8] Nisan N. Roughgarden T. Tardos E. Vazirani V.V. *Algorithmic Game Theory*. Cambridge University Press, 2007.
[9] Stephens D. W. Cumulative benefit games: Achieving cooperation when players discount the future. *J. Theoretical Biology*, 205(1):1–16, 2000.

[1] Department Of Computer Science, Babes-Bolyai University, Kogalniceanu 1, 400084 Cluj-Napoca, Romania
E-mail address: brsi0024@scs.ubbcluj.ro

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 151–154

BLOG ZEITGEIST

MIHAI DAN NADĂŞ[1]

Abstract. One of the communication phenomena that the Internet has enabled in the last years is blogging. Taking into consideration the sociological effects that are developing in the blogosphere a few questions arise, such as: how is the blogosphere evolving, do events that develop in the blogosphere influence the real world or how could one efficiently analyze real time blog data? This paper addresses the first question just after it introduces the reader into the notions that represent the foundation of all the results presented. Thus it will define the concept of blog, blogosphere, blog and post networks and cascades. At the end the cascade development model will be described along with the conclusions and future work.

1. Introduction

One of the observations from which this research starts from is that the events that happen in the blogosphere sometimes affect the real world. For example if one blogger complains about the poor quality of his recently acquired car and his experience is cited by other bloggers a decrease in the manufacturer's sales for that model could appear. The problem is that at the moment of writing this paper there is no way of determining if a particular blogosphere event will affect the real world and how. The ultimate objective of this research is to find a model that binds blogosphere events to real world events. In order to achieve this, a blog dataset from ICWSM that contains all the blog posts recorded from August 1st 2008 until September 30th 2008 was analyzed, consisting of aproximately 44 million posts [4].

Blogs were studied in favor of other Internet social environments (e.g. discussion forums, chat panels) because within the blogosphere there are important properties that can be used to monitor the events that happen inside it, the most notable are the conversational cascades.

2. Preliminaries

A blog (a contraction of the term "Web log") is a Web site, usually maintained by an individual with regular entries of commentary, descriptions of events, or other material such as graphics or video. Entries are commonly displayed in reverse-chronological order [5].

2000 *Mathematics Subject Classification.* 91C99.
Key words and phrases. blogosphere, social media, complex networks, sentiment analysis.

The blogosphere represents the collection of all the blogs together with the relations between them. The most common feature of the blogosphere is the web inheritance of links. Through links, blogs can become interconnected leading to a network with interesting properties.

Blogs are composed of posts and each post can contain several links to other blogs (or other websites). Because the paper addresses the analysis of the blogosphere only the links that refer other blogs were taken into account. From these entities two networks were constructed: a blog network and a post network.

The blog network is a directed weighted graph where a vertex is the representation of a blog and a weighted arc is the representation of all the links that connect two vertices, weighing the sum of all the links directed to the source vertex. Alternately, the post network is also a directed weighted graph where a vertex represents a post and an arc is a link that connects two posts. In order to easily take time into consideration, the weight of an arc is represented by a value Δ, where $\Delta = t_{p2} - t_{p1}$, meaning the time that passed between the moment post 1 was published and the time in which post 2 referred post 1.

The cascade concept is the foundation for the discoveries presented in this paper and for the major future work. A cascade is a subgraph of the post network, formed from all the descendents of a certain root post. Cascades in this context can be easily regarded as conversational graphs.

The reason for which cascades may represent a high relevancy through the blogosphere properties is that they actually reveal the level of interest a certain post has in the micro-society it is available in.

3. Cascade Topology

The blogosphere is known being a complex network [3]. This easily takes the discussion to scale-free theory. A network is named scale-free if its degree distribution, i.e., the probability that a node selected uniformly at random has a certain number of links (degree), follows a particular mathematical function called a power law [1].

The cascades that develop in the blogosphere basically point out the degree of interest that the original article generated. This property can be used to track down influential events that may have reached the real world.

In order to achieve some kind of prediction a model for the dynamics of the blogosphere had to be determined based on the properties of the analysed data and it's cascades. Figure 1 shows the results of the cascade analysis conducted.

4. The model

The following model represents the innovation and originality of this paper. It basically manages to provide a very precise way of generating blogosphere cascades, representing an important step into achieving the proposed goal.

The model, entitled Blogosphere Topology Model (BTM) is inspired from the SIS (Susceptible-Infected-Susceptible) [2], used in epidemiology. The model handles the network agents which can have two states: susceptible and infected. When an agent is infected it can pass its state (i.e. infected) to any of its descendants (that

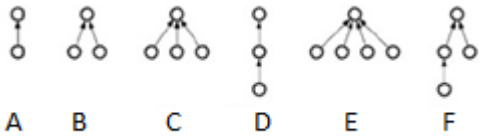


FIGURE 1. Frequency topological chart. Starting from left to right we may see the shape of the cascades starting from the most frequent one and ending with more complex scenarios. There were found a total of 1.1M cascades, from which 97% are trivial (isolated posts), 2.3% are simple (first row from the figure) and the rest are rather complex. One observation that can be made here is that cascades tend to be wide and not too deep.

are susceptible) with a fixed probability. Once an agent infects a neighbor his state becomes automatically susceptible, thus leaving open the possibility of creating cycles in the network.

In the case of the blogosphere the agents described in the model are the actual blog posts. In order to generate a cascade the only action that has to be handled is to select randomly a blog post from the blogosphere's post network, set its state to infected and then infect the incident posts with a certain probability.

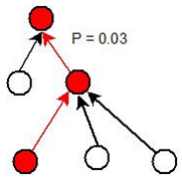


FIGURE 2. The BTM simulating the building of a cascade in the post network. The simulation starts by randomly choosing a node from the network and assigning it the infected state. With a probability of 0.03% the node infects adjecent nodes and upon infection it becomes susceptible again. The process repeats as long as new nodes are infected.

The reason for which this model is appropriate for simulating the dynamics of blog cascades is that it maintains the topological and mathematical features of the cascades as seen in the previous observations. Tests have shown that with the probability of infection set to 0.03% the properties of the generated cascades have been very similar to the previously infered ones.

Cascade Topology Model	Real Data Occurance	Generated Occurence
A	84.3%	76.6%
B	9.1%	10.3%
C	3.7%	3.5%
D	1.2%	2.4%
Other	1.7%	7.2%

TABLE 1. The table presents the comparative data between the topology of the cascades found in the real data and those generated by the proposed model. It is clear enough that the resemblence is very good and that the model behaves well on generating real-world cascades.

5. Conclusions & Future Work

Applying advanced analysis methods on the blogosphere is a relatively new domain and this could lead to very interesting results. Currently the focus is to provide an optimized analysis environment in order to properly isolate blogosphere events and find real world consequences generated by them. Once such an environment is complete and a binding model is optimized, the findings could easily be applied in several business scenarios starting from sales, marketing and ending with management and business intelligence needs.

References

[1] A. Barabasi, E. Bonabeu, *Scale-Free Networks*, Scientific American, May 2003, 50-59
[2] T. Ichinomya, *Analysis of Susceptible-Infected-Susceptible Model on Complex Network*, `www.topology.coe.hokudai.ac.jp`
[3] X. Shi, B. Tseng, L. A. Adamic, *Loooking at the Blogosphere Topology Through Different Lenses*, Proceedings of the ICWSM, 2007
[4] ***, *International AAAI Conference on Weblogs and Social Media*, `http://www.icwsm.org`
[5] ***, *State of the Blogosphere 2008*, Technorati, `http://technorati.com/blogging/state-of-the-blogosphere`

[1] Faculty of Mathematics and Computer Science, Babeş-Bolyai University, Cluj-Napoca, Romania
E-mail address: `mihai@nadas.ro`

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 155–158

A SOFTWARE TOOL FOR INTERACTIVE DATABASE ACCESS USING CONCEPTUAL GRAPHS

VIORICA VARGA[(1)], CHRISTIAN SĂCĂREA[(1)], AND ANDREA TAKACS [(1)]

Abstract. The basic idea of using Conceptual Graphs as query interface to relational databases has been stated by Sowa. The full expressiveness of modern database query languages includes negation and the so-called aggregating functions too. These features were introduced in [1] by means of Nested Concept Graphs with Cuts. The novelty of the paper is the software named CGDBInterface, which offers a graphical tool to query an existing relational database. The aim of our software tool is to connect to an existing database by giving the type and the name of the database, a login name and password, then the software offers the structure of the database in form of a conceptual graph. The software provides a graph editor and a wizard to assist the user in constructing a query with Nested Concept Graphs with Cuts. To the best of our knowledge, such a software was not developed yet.

1. Mathematical background

R. Wille described in [4] how Conceptual Graphs and Formal Concept Analysis can be connected through their conceptual structures. An *abstract concept graph* is defined as a mathematical structure $\mathfrak{G} := (V, E, \nu, C, \kappa, \theta)$ for which

(1) V and E are finite sets and ν is a mapping of E to $\bigcup_{i=1}^{n} V^i (n \geq 2)$ so that (V, E, ν) can be considered as a finite directed multi-hypergraph with vertices from V and edges from E (where $e \in E$ and $|e| := k \Leftrightarrow \nu(e) = (v_1, \ldots, v_k)$),
(2) C is a finite set and κ is a mapping of $V \cup E$ to C such that $\kappa(e_1) = \kappa(e_2)$ always implies $|e_1| = |e_2|$,
(3) θ is an equivalence relation on V.

In order to describe the conceptual structure of a conceptual graph as a conceptual hierarchy, a *power context family* is defined as $\mathbb{K} := (\mathbb{K}_1, \ldots, \mathbb{K}_n)(n \geq 2)$ with $\mathbb{K}_j := (G_j, M_j, I_j)(j = 1, \ldots, n)$ such that $G_j \subseteq (G_1)^j$. An *abstract concept graph over the power context family* $\mathbb{K}$ is an abstract concept graph $\mathfrak{G} := (V, E, \nu, C, \kappa, \theta)$, with $C = \bigcup_{i=1}^{n} \underline{\mathfrak{B}}(\mathbb{K}_i)$, $\kappa(V) \subseteq \underline{\mathfrak{B}}(\mathbb{K}_1)$, and $\kappa(e) \in \underline{\mathfrak{B}}(\mathbb{K}_i)$ for all $e \in E$ with $|e| = i$.

In order to connect with previous work ([3]), for a given relational database the CGDBInterface tool will also compute the *realized concept graph* of a convenient

2000 *Mathematics Subject Classification.* 68P15 Database theory, 03G10 Lattices and related structures.

Key words and phrases. Conceptual Graphs, Interactive Database Access.

power context family corresponding, i.e., the mathematical structure describing the conceptual structure of the conceptual graphs of a relational database by means of Formal Concept Analysis.

In order to describe negation and aggregating functions, we need to implement the so-called *nested concept graphs with cuts* ([1]).

A *nested relational graph with cuts* is a structure $(V, E, \nu, \top, Cut, area)$, where

(1) V, E, and Cut are pairwise disjoint, finite sets whose elements are called *vertices, edges*, and *cuts*, respectively,
(2) $\nu\colon E \to \bigcup_{k\in\mathbb{N}} V^k$ is a mapping,
(3) $\top$ is a single element called the *sheet of assertion*, and
(4) $area\colon V \cup E \cup \{\top\} \to \mathfrak{P}(V \cup E \cup Cut)$ is a mapping such that
   (a) $area(k_1) \cap area(k_2) \neq \emptyset \Rightarrow k_1 = k_2$ for $k_1, k_2 \in V \cup Cut \cup \{\top\}$,
   (b) $V \cup E \cup Cut = \bigcup\{area(k) \mid k \in V \cup Cut \cup \{\top\}\}$,
   (c) $x \notin area^n(x)$ for each $x \in V \cup \{\top\} \cup Cut$ and $n \in \mathbb{N}$ (with $area^0(x) := \{x\}$ and $area^{n+1}(x) := \bigcup\{area(y) \mid y \in area^n(x)\}$).

Nested concept graphs with cuts are constructed from nested relational graphs with cuts by additionally labelling the vertices and edges with names.

2. Software description

This section presents how our software constructs both the conceptual graph corresponding to an existing relational database and the corresponding concept graph over a power context family. The actual version of application CGDBInterface can connect to MS SQL Server, Oracle and mySQL databases. Having a valid user name and password the user can see in the first screen the conceptual graph of the relational database. As defined in [2], the table names and their attributes become concepts. The attributes of a table are linked by conceptual relations to the concept corresponding to the table they belong to. The relationships between tables are designed by conceptual relations too. To the best of our knowledge, this is a novelty of our approach. The software reads from the system tables of the database the name of the tables and their attribute names. In the relational data model, the relationship between tables are modeled by foreign key constraints. The foreign key constraints are retrieved from system catalogs too. In order to draw the conceptual graph corresponding to the chosen database Graph Visualization Software (Graphviz) is used. A database may have a very large number of tables; every table may have a very large number of attributes. In order to fit in one screen, we visualize on the first step only the table names and the relations between them.

Example 1. Let be the next relational database table scheme:

```
Department [departmentID, name]
Teacher [teacherID, firstName, lastName, departmentID]
Specialization [specializationID, name, language]
Group [groupID, specializationID]
Student [SSN, firstName, lastName, GroupID, cgpa]
Course [courseID, name, teacherID]
```

```
Attends[SSN, courseID, mark]
```

The software provides to enlarge the view, so that the attributes will be visible, see Figure 1.

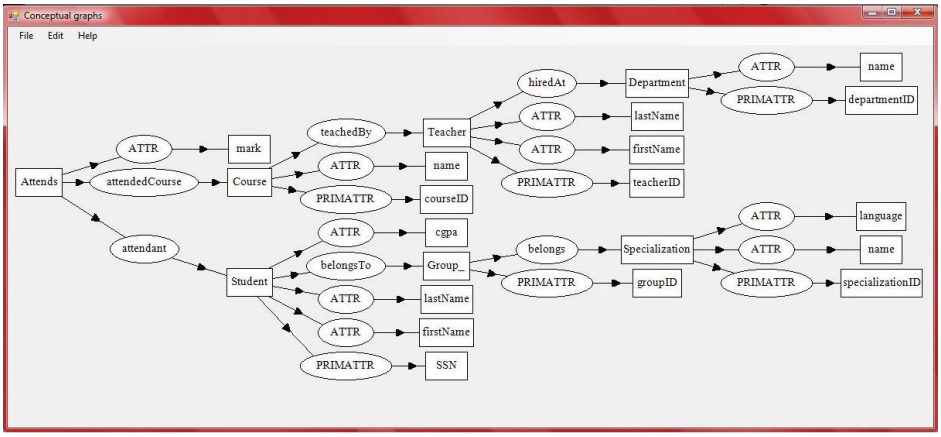


FIGURE 1. Conceptual graph interface for Faculty database in detail

In order to query the database, the CGDBInterface tool offers a graph editor and a wizard. The graph editor provides rectangles for modeling concepts, circles for conceptual relations and arrows to link them. The wizard offers the conceptual graph of the database. The user can select tables from it by clicking on concepts. The wizard give the possibility to construct queries with aggregate functions and negation too. These features are implemented using Nested Concept Graphs with Cuts. After the user constructs the query graph the software will generate the SELECT statement and execute it against the database.

Example 2. Let be the next SELECT statement.

```
SELECT firstname, lastname FROM Student WHERE cgpa = 10
```

In Figure 2 (a) we can see the query graph which was constructed with our software.

Example 3. A more complicated SELECT statement is:

```
SELECT s.lastName, s.firstName
FROM Student s, Attends a, Course c
WHERE s.SSN = a.SSN and a.courseID = c.courseID
    and c.name = "Conceptual graphs"
```

The Figure 2 (b) depicts the corresponding query graph.

3. CONCLUSION

Interaction with computers becomes more and more important in our daily lifes. The goal of conceptual graphs is to provide a graphical representation for logic which is able to support human reasoning. This article proposes an application which provides a graphical interface for database interaction in form of conceptual graphs.

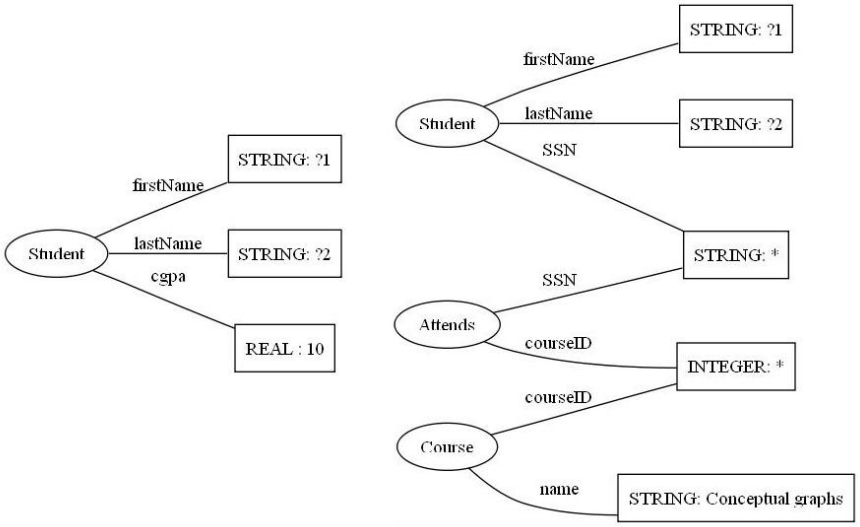


FIGURE 2. (a) Simple query (b) Query involving join operation

REFERENCES

[1] Dau, F., Hereth, J. C.: Nested Concept Graphs: Mathematical Foundations and Applications for Databases. In: Ganter, B.; de Moor, A. (eds.): Using Conceptual Structures. Contributions to ICCS 2003. Shaker Verlag, Aachen, (2003), pp. 125-139.

[2] Guy, M. W., Moulin B., Sowa, J. F. eds.: Conceptual Graphs for Knowledge Representation. Lecture Notes in AI 699, Springer-Verlag, Berlin (1993).

[3] Janosi Rancz, K.T., Varga, V., Puskas, J.: A Software Tool for Data Analysis Based on Formal Concept Analysis, Studia Univ. Babeş-Bolyai, Informatica, 53, 2 (2008), pp. 67-78.

[4] Wille, R.: Conceptual Graphs and Formal Concept Analysis, Lecture Notes In Computer Science; Vol. 1257, Proceedings of the Fifth International Conference on Conceptual Structures: Fulfilling Peirce's Dream, Springer Verlag (1997), pp. 290-303.

(1) BABES-BOLYAI UNIVERSITY CLUJ NAPOCA, STR. KOGALNICEANU 1
E-mail address: ivarga@cs.ubbcluj.ro, csacarea@math.ubbcluj.ro, taim0694@scs.ubbcluj.ro

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 159–162

SEMI-SUPERVISED FEATURE SELECTION WITH SVMS

ZALÁN BODÓ AND ZSOLT MINIER[1]

Abstract. Feature selection plays and important role in machine learning: eliminates irrelevant dimensions thus turning the learner into a better, more efficient system. In this paper we use non-linear semi-supervised SVMs for feature selection and through experiments we demonstrate the efficiency of the methods, showing how unlabeled data can lead to a better reduction. Semi-supervised feature selection is achieved by using semi-supervised/cluster kernels, that is embedding the information provided by the unlabeled data into the kernel, and applying dimensionality reduction methods developed for non-linear SVMs.

1. Introduction

Semi-supervised learning (SSL) is a special case of classification; it is halfway between classification and clustering. In SSL the training data is augmented by a set of unlabeled data samples, that is we have $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_\ell, y_\ell)\} \cup \{\mathbf{x}_{\ell+1}, \mathbf{x}_{\ell+2}, \ldots, \mathbf{x}_{N:=\ell+u}\}$, where usually there are far less labeled data than unlabeled ones, i.e. $\ell \ll u$. Semi-supervised learning is the problem of assigning labels to the unlabeled samples of the data set using the information provided by both the labeled and the unlabeled data.

Feature selection or dimensionality reduction methods are important for machine learning algorithms, because many problems usually deal with thousands of features, some of them representing only noise, others being strongly correlated, etc. Therefore in order to handle this high dimensionality and build efficient learners the elimination of irrelevant features is needed. Feature selection methods can be classified as follows [4]: wrappers, embedded methods and filters. Wrappers use an arbitrary machine learning technique with some heuristics for choosing the best feature set observing the performance of the classifier for these feature sets. Embedded methods are implicitly built into the learning technique, i.e. learning the optimal decision function and finding the best feature subset are performed simultaneously. Perhaps the most successful feature selection methods are filters, which choose a subset of features according to a particular measure.

The most popular feature selection techniques are either supervised or unsupervised methods. We expect that by using semi-supervised learners (SVMs) for feature

2000 *Mathematics Subject Classification.* 68T05, 68T30.
Key words and phrases. semi-supervised learning, feature selection, kernel methods.

Algorithm 1 SVM-based feature selection using backward elimination

1: $F = \{1, 2, \ldots, d\}$
2: **repeat**
3: Train an SVM (using a semi-supervised/cluster kernel) with all the training data using the features from F
4: **for** $f_i \in F$ **do**
5: Evaluate the ranking criterion $R_c(f_i)$
6: **end for**
7: $f_r = \text{argmin}_i R_c$ ▷ Determine most irrelevant dimension
8: $F = F \setminus \{f_r\}$ ▷ Remove most irrelevant dimension
9: **until** $F = \emptyset$

selection we can achieve a better dimensionality reduction, that is the larger unlabeled data set induces a better decision boundary and thus a smaller feature set.

2. Semi-supervised support vector machines

The Laplacian support vector machine [1] approaches the semi-supervised learning problem by introducing an additional regularization term involving the graph Laplacian $\mathbf{L}$, that reflects the intrinsic geometry of the data:

$$\min F(\mathbf{w}, b, \boldsymbol{\xi}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{\ell}\xi_i + \gamma_I \mathbf{f}'\mathbf{L}\mathbf{f}$$
$$\text{such that} \quad y_i(\mathbf{w}'\phi(\mathbf{x}_i) + b) \geq 1 - \xi_i,\ \xi_i \geq 0, \quad i = 1, \ldots, \ell$$

where $\mathbf{f} = [f(\mathbf{x}_1) \ldots f(\mathbf{x}_N)]'$. Interestingly this can be solved by solving the supervised SVM problem using the kernel [2]

$$\tilde{k}(\mathbf{x}_1, \mathbf{x}_2) = k(\mathbf{x}_1, \mathbf{x}_2) - \mathbf{k}'_{\mathbf{x}_1}(\frac{1}{4\gamma_I}\mathbf{I} + \mathbf{L}\mathbf{K})^{-1}\mathbf{L}\mathbf{k}_{\mathbf{x}_2}$$

Other possibilities to use SVMs for semi-supervised learning exist. One example is to build a kernel based on the information provided by the labeled and unlabeled data, that is to use the unlabeled data to build a new representation of the samples [2], then train SVMs in a supervised setting with this kernel.

3. Semi-supervised feature selection

SVMs and linear classifiers in general, $\mathbf{w}'\mathbf{x}$, can be used for feature selection by eliminating those dimensions for which the magnitude of w_i is small. The feature selection method proposed in [5] ranks the dimensions by w_i and eliminates one or more features with small rank. This can be argued by the simple fact that large magnitude dimensions contribute more to the final decision. However this works only in the input space; introducing kernels for non-linear decision functions this selection should be done in the feature space, which could be of infinite dimensionality, therefore no straightforward solution is possible.

For eliminating the dimension with the smallest influence on the predicted label in the non-linear case, we calculate the "ranking criterion" as the variation of $\|\mathbf{w}\|^2$ caused by the removal [5]:

$$\left|\|\mathbf{w}^2 - \|\mathbf{w}^{(i)}\|^2\right| = \left|\boldsymbol{\beta}'\mathbf{K}\boldsymbol{\beta} - \boldsymbol{\beta}^{(i)\prime}\mathbf{K}^{(i)}\boldsymbol{\beta}^{(i)}\right| \tag{1}$$

where $\boldsymbol{\beta}_j = \alpha_j y_j$, $\mathbf{K}^{(i)}$ denotes the kernel matrix after removing the ith feature, and $\boldsymbol{\beta}^{(i)}$ denotes the vector corresponding to the solution $\boldsymbol{\alpha}^{(i)}$. This approach is called the SVM-RFE algorithm. For complexity reasons usually $\boldsymbol{\beta}$ is used instead of $\boldsymbol{\beta}^{(i)}$. In the experiments both approaches are tested. Other methods include using leave-one-out error (L) bounds for SVMs for the ranking criterion, like the radius-margin bound of Vapnik [6]. Since we are using L1-SVMs we apply a radius-margin bound proposed in [3]:

$$L \leq \left(R^2 + \frac{1}{C}\right)\left(\|\mathbf{w}\|^2 + C\sum_{i=1}^{\ell}\xi_i\right) \tag{2}$$

where R denotes the radius of the smallest sphere containing the data $\phi(\mathbf{x}_i)$ in the feature space. Using a backward elimination scheme the expression from eq. (1) and the right hand side of eq. (2) can be directly used as a ranking criterion for feature elimination: the feature which minimizes the ranking criterion is removed from the feature set. The scheme of the feature selection algorithm used by us is shown in Algorithm 1 [5, 6].

4. Experiments and discussion

In our experiments we tested the methods on a non-linear synthetic data set described in [7]. From the generated 52 dimensions only 2 are relevant. These dimensions are constructed as follows: if $y = -1$ then the two dimensions are drawn from $N(\boldsymbol{\mu}_1, \boldsymbol{\Sigma})$ or $N(\mu_2, \boldsymbol{\Sigma})$ with equal probability, $\boldsymbol{\mu}_1 = [-3/4, -3]'$, $\boldsymbol{\mu}_2 = [3/4, 3]'$, $\boldsymbol{\Sigma} = \mathbf{I}$, if $y = 1$ then the dimensions are drawn again from two normal distributions with equal probability having the parameters $\boldsymbol{\mu}_1 = [3, -3]'$, $\boldsymbol{\mu}_2 = [-3, 3]'$, and $\boldsymbol{\Sigma} = \mathbf{I}$. The remaining 50 features are noise, each one is generated from the normal distribution $N(0, 20)$.

Table 1 shows the obtained results. Because of lack of space the complete settings of the test are not described here. SVM-RFE and SVM-RMB denote the methods using equations (1) and (2) with supervised SVMs, LapSVM-RFE and LapSVM-RMB denote that a Laplacian SVM was used for learning, while the ^{r} at the end means that retraining was performed.

The results show that feature selection methods using SVMs clearly outperform LapSVMs on the synthetic non-linear data set used. Only the results of LapSVM-RMB^r with 100 points can be considered as acceptable. The methods require further tests on data sets where semi-supervised assumptions hold, and detailed analysis of the results obtained.

Methods	Training set size			
	10	**30**	**50**	**100**
SVM-RFE	$47.06 \pm 8\%$	$25.57 \pm 20\%$	$9.63 \pm 12\%$	$5.78 \pm 3\%$
SVM-RFE^r	$49.78 \pm 5\%$	$32.47 \pm 19\%$	$24.48 \pm 19\%$	$18.72 \pm 18\%$
SVM-RMB	$51.02 \pm 4\%$	$46.85 \pm 8\%$	$50.67 \pm 5\%$	$47.28 \pm 8\%$
SVM-RMB^r	$48.52 \pm 6\%$	$36.30 \pm 18\%$	$14.07 \pm 16\%$	$6.65 \pm 8\%$
LapSVM-RFE	$48.47 \pm 6\%$	$43.83 \pm 13\%$	$43.27 \pm 13\%$	$48.50 \pm 7\%$
LapSVM-RFE^r	$48.35 \pm 10\%$	$46.95 \pm 9\%$	$41.40 \pm 14\%$	$41.27 \pm 15\%$
LapSVM-RMB	$48.80 \pm 6\%$	$44.30 \pm 12\%$	$37.95 \pm 15\%$	$30.57 \pm 19\%$
LapSVM-RMB^r	$49.43 \pm 4\%$	$40.57 \pm 13\%$	$28.25 \pm 16\%$	$7.13 \pm 9\%$

TABLE 1. Test errors (mean and standard deviation) for the non-linear synthetic problem using SVMs and Laplacian SVMs for feature selection.

REFERENCES

[1] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7:2399–2434, 2006.

[2] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. *Semi-Supervised Learning*. MIT Press, September 2006.

[3] Kai-Min Chung, Wei-Chun Kao, Chia-Liang Sun, Li-Lun Wang, and Chih-Jen Lin. Radius margin bounds for support vector machines with the RBF kernel. *Neural Computation*, 15(11):2643–2681, November 2003.

[4] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

[5] Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik. Gene selection for cancer classification using support vector machines. *MACHLEARN: Machine Learning*, 46, 2002.

[6] Alain Rakotomamonjy. Variable selection using SVM-based criteria. *Journal of Machine Learning Research*, 3:1357–1370, 2003.

[7] Jason Weston, Sayan Mukherjee, Olivier Chapelle, Massimiliano Pontil, Tomaso Poggio, and Vladimir Vapnik. Feature selection for SVMs. In Todd K. Leen, Thomas G. Dietterich, and Volker Tresp, editors, *NIPS*, pages 668–674. MIT Press, 2000.

[1] FACULTY OF MATHEMATICS AND COMPUTER SCIENCE, BABEŞ–BOLYAI UNIVERSITY, CLUJ-NAPOCA

E-mail address: {zbodo, minier}@cs.ubbcluj.ro

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 163–166

SENSITIVE ANTS ALGORITHM FOR ROUTING IN TELECOMMUNICATION NETWORKS

ALEXANDRA-ROXANA TĂNASE[(1)]

Abstract. Ants with a sensitive reaction to pheromone are considered to be multi-agents for the metaheuristic called *Sensitive ACS (SACS)*. In the *SACS* model, each ant is endowed with a pheromone sensitivity level which allows different types of responses to pheromone trails. Such an artificial system, controlled by emergent behavior promise to generate engineering solutions to distributed systems management problems found, for example, in telecommunication networks. A comparison between AntNet Algorithm (based on ACO model) and Sensitive Ants Agorithm (based on SACS model) is made, for solving routing problems.

1. Introduction

Ant as a part of a well organised colony, can be a powerful agent, working for the development of the colony. Each ant is able to communicate, learn, cooperate, and all together they are capable to develop themselves and colonise a large area. *Stigmergy* is defined as a method of indirect communication in a self-organizing emergent system where its individual parts communicate with one another by modifying their local environment. In many ant species, ants walking from or to a food source, deposit on the ground a substance called *pheromone*.

On sensitive ants we may deal with a degree of smelling the pheromone. This is called: Pheromone Sensitivity Level (PSL), and has a value between 0 and 1 [2,3,4]. The idea of using ants for solving routing problems is not new; for instance, the ACO metaheuristic provides good results in this area [5]. The paper aim is to provide an algorithm where sensitive ants can be used for routing. Numerical experiments indicate the potential of the proposed algorithm.

2. Sensitive Ants model for routing in telecommunication networks

Routing can be characterized by the following general way. Let the network be represented in terms of a directed, weighted graph: $G = (V, E)$, where each node in the set V represents a processing and forwarding unit and each edge in E is a transmission system with some capacity/bandwidth and propagation characteristics. The node from where the traffic flow originates is also called source, while the nodes to

2000 *Mathematics Subject Classification.* 90C59, 90B18.
Key words and phrases. metaheuristic, sensitivity, communication networks.

which traffic is directed are the final end-points, or destinations. The nodes between sources and destinations are called intermediate or relay nodes [1,5,6].

The characteristics of the routing problem make it well suited to be solved by a mobile multi-agent approach. The ideea of using ants in routing problems is not new, i.e. M. Dorigo and G. Di Caro, originally proposed four ACO algorithms for adaptive agent-based routing. They are the following: AntNet, and some improvements of it: AntNet-FA, AntNetSELA and AntHoc Net [5,6,7].

The aim of the *Sensitive Ants Algorithm (SAA)* is to provide the best route using ants with random or global (fixed) pheromone sensitivity. The algorithm refers to finding a minimum-cost path from a certain source to a randomly chosen destination, using ants. It stops when the best path is achieved and all the updates are made in the routing tables. The routing algorithm uses two classes of ants: first one is the management ants; here three types of ants are considered: exploration ants, message ants or response ants (they work as backward ants) and error ants. The second class are exploitation ants which only take into account the improvements made by management ants. They are also called data packets while the management ants may be called routing packets [5,6]. The SAA partially follows the SACS model. In SAA metaheuristic, transition probability is expressed as:

$$p_{iu}^{k} = \frac{[\tau_{iu}(t)]\cdot\eta_{iu}(t)^{PSL}}{\sum_{o\in J_i^k}[\tau_{io}(t)]\cdot\eta_{io}(t)^{PSL}}.$$

The formula above expresses the probability of ant k, from the node i, to choose the next node u; PSL is the pheromone sensitivity of the ant; J_i^k are the unvisited neighbors of the node i; η is the visibility value: $\eta = \frac{1}{c_{ij}}$; where c_{ij} is the cost between node i and node j while in SACS metaheuristic it is expressed as:

$$p_{iu}^{k} = \frac{[\tau_{iu}(t)]\cdot\eta_{iu}(t)^{\beta}}{\sum_{o\in J_i^k}[\tau_{io}(t)]\cdot\eta_{io}(t)^{\beta}},$$

where β is a parameter used for tuning the relative importance of edge cost in selecting the next node. The algorithm can be resumed as follows: from each network node, ants are randomly launched towards specific destination nodes. The agent generation processes happen concurrently and asyncronously. The agents moving from their source to the destination node are called forward ants, and they can be identified by the following formula: $F_{s\to d}$ [5]. At iteration $t+1$ every ant moves to a new node and the parameters controlling the algorithm are updated. At each time unit evaporation takes place. The pheromone on the trail is updated as follows:

$$\tau_{i,j}(t+1) = (1-\rho)\cdot\tau_{i,j}(t) + \ln(N_{i,j}(t)+1),$$

where $N_{i,j}$ is the number of ants which pass on the edge $i \to j$ at iteration t ; ρ represents evaporation rate. If an ant arrives at destination, it is deleted and a backward ant (response ant), $B_{d\to s}$, is created and goes back following the same path $P_{s\to d} = [s, v_1 v_2 \ldots, d]$ as before but in the opposite direction.

3. Numerical experiments

SAA paradigm presents a simulation of a routing network, using a network which is represented by an n-node undirected graph $G = (V, E)$. NFSNet, one of the graphs used for the simulation, is a WAN composed of 14 nodes and 21 bi-directional links

with a bandwidth of 1.5 Mbit/s [5]. SAA showed good performance under the NSFNet graph. All reported data are averaged over 10 trials. The best results were rather obtained using sensitive ants, with a random PSL, than ants with a global PSL. Costs on the edges were computed taking into account only the propagation delays: costs range from 4 to 20 msec. Time To Live (TTL) for an ant (packet) was set at 255 sec.

TABLE 1. Results on ants with different PSL

PSL	No of Steps	No of Ants	No of Packets	No of Delivered packets
random	**2524.8**	**274.5**	**277.8**	**255.4**
0.2	2653.2	291	290.7	266.9
0.5	2570.8	286.6	279.7	257.9
0.7	2573.7	281.1	277.9	255
1	2702.1	299	300.5	277.8

TABLE 2. Time results of SAA on NSFNet

PSL	No of Steps	Time (sec)
random	**2524.8**	**37.8**
0.2	2653.2	39.7
0.5	2570.8	38.5
0.7	2573.7	38.6
1	2702.1	40.5

As it can be seen from Table 1, the better results were obtained using a random PSL value, so sensitive ants (with random PSL values between 0 and 1) are better than ants which have a global PSL value, i.e., $0,2$, $0,5$, 0.7 or 1. In ACO metaheuristic, every ant has the same global value for the PSL and this is considered to be 1. Time of the simulation is proportional with the number of steps; a step occurs at every 15 msec. The number of ants and the number of packets generated during the simulation, represent the neccesarry values for the algorithm to reach the best path in every situation presented above. All the results from the tables represent average values of the data.

TABLE 3. The percentage of delivered packets on SAA vs AntNet

Application	Percentage of delivered packets
SAA	92%
AntNet	90%

4. Conclusions

The Sensitive Ants Algorithm based on the model SACS is presented. SAA obtained good results in routing taking into account the NSFNet graph which was also used by M. Dorigo et al. in AntNet paradigm. The computational results concerning the SAA model show that sensitive ants achieve better results than ants with global PSL because exploration of the map can be made better with ants which have a random PSL. This results may be improved by considering different parameter settings.

References

[1] Baran B., Sosa R.: AntNet: Routing Algorithm for Data Networks based on Mobile Agents; a DIPRI Research Grant of the National University of Asuncion-Paraguai, (2000)

[2] Chira C., Pintea C.-M., Dumitrescu D.: Sensitive Stigmergic Agent Systems, Adaptive and Learning Agents and Multi-Agent Systems (ALAMAS), Maastricht, The Netherlands, MICC Technical Report Series, number 07-04, ISSN 0922-8721, pp. 51-57, (2007)

[3] Chira C., Pintea C.-M., Dumitrescu D.: Sensitive Ants: Inducing Diversity in the Colony; International Workshop on Nature Inspired Cooperated Strategies for Optimization, NICSO, Puerto de la Cruz, Tenerife, (2008)

[4] Chira C., Pintea C.-M., Dumitrescu D.: Sensitive Ant Systems in combinatorial optimization; Proceedings of the International Conference on Knowledge Engineering, Principles and Techniques, KEPT2007, Cluj-Napoca (Romania), pp. 185-192, (2007)

[5] Di Caro G.: Ant Colony Optimization and its Application to Adaptive Routing in Telecommunication Networks, Ph.Dissertation, Bruxelles, (2004)

[6] Di Caro G., Dorigo M.: AntNet: Distributed Stigmergetic Control for Communications Networks; IRIDIA- Journal of Artificial Intelligence Research 9 317-365, (1998)

[7] Dorigo M., Birattari M., Stützle T.: Ant Colony Optimization Artificial Ants as a Computational Intelligence Technique, Published by: IRIDIA-Technical Report Series ISSN 1781-3794, (2006)

[1] Faculty of Mathematics and Computer Science, Babes-Bolyai University, Kogalniceanu 1, RO-400084 Cluj-Napoca, Romania

E-mail address: tasi0034@scs.ubbcluj.ro

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 167–170

EMERGENCY SERVICE SYSTEMS AND ROBOTS

ALINA MIRON[1]

Abstract. Emergency service systems are evolving once with the development of different technologies. The tendency of the emergency service systems is to become universal services that can deal with a great variety of situations. Steps are made towards the introduction of robots in such systems but there exist many questions that come into view: how do robots search for people, should there be used independent robots or controlled from distance, when and how the decision to announce an emergency service is made. We will try to answer this questions and focus our attention on the integration of robots in the emergency service system, presenting some experimental results obtained using a robot Robotino, a couple of PDAs and a laptop. This experiment main disadvantance is the poor precision of the human finding process but this is a problem to which we'll try to present a solution.

An emergency service is an organization which ensures public safety by addressing different emergencies. Any of these services could be enhanced by the use of robots. This is necessary because if you take into consideration the case of a natural disaster the emergency systems on their own will not be able to handle the situation in a short time. Here time will mean life. In this case robots could be used to search for people, such that in the moment a person is found they will announce this to a server and transmit its position.

1. Robots in emergency situations

Robots can have many usages in emergency situations: from reaching places human could not (the use of robotic technology allows people to remotely explore and work in dangerous environments without the risk of injury) to finding people. Applications of robots to various emergency response can include urban search and rescue, bombdisposal or other explosive, hazardous material inspection and cleanup.

Emergency response robots must have different characteristics accordingly to the way they were design to response, but common areas of technologies could be share. From the physical point of view emergency response robots must have mechanical durability, high mobility in different environments, manipulation capabilities, recovery ability from errors or from fallings.

Should a robot be independent? In most situations of emergency a robot must act independently because this can make the searching process faster. But for now it is

2000 *Mathematics Subject Classification.* 68T45, 68T40.
Key words and phrases. robots, emergency systems, human detection.

very difficult to construct completely independent robots due to the lack of precision (poor sensors, rigid algorithms), and because this can not be afforded in an emergency situation human assistance is necessary in a certain amount - for example to indicate to the robot the area to be covered by the searching process, or to confirm if a find is endeed a human or not.

How can a robot search for humans? The process of human search can be a very difficult one. For example if you rely on the vision sensor - for example a web cam, the difficulty is in deciding if a certain form can be a human or not. The human body is highly complex, and because of the many possitions it can take, different cloths, different colors,it's making the search based on the computer vision a highly complex one. It's worth noting that nearly all the computer vision algorithms so far deal with face detection problems and a few with pedestrian detection, that is the hole human body can be seen and it has a certain position. In what will follow it will be proposed an algorithm for human detection in more complex situations, for example when the human is laid on the ground, and the robot must decide whether the found object is endeed a human or not.

2. Scenario and experiment

There exist some services dedicated in urban search across the world whose scope is to improve disaster readiness and response through public awareness, collaboration, research and engineering, which imagined scenarios like pandemics, natural disasters or terrorist attacks [3].

Imagine a situation in which a robot equipped with a video cam (or for better results a video cam with infrared thermal images) searches for people and in the moment a person is found it announces the clients connected to it (like the PDAs of the members of the searching team) or a central server. The clients can receive raw images (such that they can perform processing of the received data), or processed images (with correction algorithms applied to the images), or better receive only news when a human is found. The connected clients are able to see the localization of the robot at any moment (localization found using a GPS system). Even if the robot can search for people in an independent mode it could also be remotely controlled.

For the beginning the robot (in the experiment a robot from the Festo company was used - Robotino) having a web cam searches for people based on an algorithm of face detection. The navigation system considered is one based on blind man strategy, in the sense that the robot does not know the area that it will have to cover and does not the obstacles it will encounter.

In the system from Figure 1 the transmitions of data between the robot and the other devices was very good on a short range (15 m), but bad on larger distances because of the chosen transmission method. The robot was able to successefuly find objects, send images with the results to the 'resque team', but unfortunately the overall performace of the system was poor - that is the purpose of finding humans was meet only in very good conditions - very good lighting, perfect alignment of the robot with the human face. This lacks can be improved through: (1) direct connection

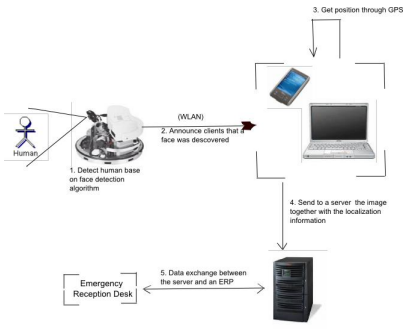


FIGURE 1. Robot system

between the robot with the devices but using as an intermidiary an internet connection and (2) design a better algorithm than the face detection one.

3. Human Detection

Like we presented above, currently the system of human detection is based on a face detection algorithm, and we presented the reasons this will not perform well in a real situation. But what ways do we have to provide a better algorithm?

Human body is a complex structure,and what makes difficult to detect a human based only on image processing techniques is the variation in pose, clothing, appearance, background, illumination etc. That is why we'll consider a more complex approach.

From the computer vision point of view, most of the algorithms, including face detection are based on 2D searches. There also exist some proposals for 3D searches [4] that uses 3D scanning grid. This method provided good results - actually better results than the 2D searches, but makes some assumptions: like the fact that people are standing, and the camera can see the entire human body. Unfortunatly in the case of an emergency, the images taken by a robot in most cases will not expose the entire body of a human, that is why we plan to extend the 3D search to parts of human body.

The moment the robot finds an object it will try to identify it. If it is a human than it will announce this to its supervizor (another device). For the algorithm of human detection it will be made through putting together several informations from the photos taken. First of all, several photos of the object will be taken - this process is an algorithm in itself because the robot must decide the number of photos to be taken, and the area that the photos cover. From each round of photos the best ones are chosen (the photos with the most clearly defined lines and edges), after which the remaining ones will be stiched together following the line of the focused object and ignoring the noise of the background. After the object is separated the 3D analysis will begin. Even in this case there are situations in which maybe the whole object (human body) will not be in the analysed image, and that is why the algorithm tries

to find the best match. First it will search for the head because this is the part of the human body that can be most easyly identified. Afterwords, based on degree of freedom it will try the reconstruction of the body based on a 3D model of the human body(finding the arms, hands, legs etc.)

Of course, if the robot could be equipted with a termal camera, the part in which it tries to identify the whole area of the found object is simplified, and then it only remains to find out if the object is or is not a human. The problem here lies in the highly cost of all this equipment that increases the cost of the system. The purpose of this system is to show that through the use of simple sensors the objective of human detection can be meet.

4. Conclusions

The field of emergency systemes based on robots is an exciting one because it can impact directly human lives. We'd wish to provide a new robot architecture based on simple sensors tailored for intervention in some emergency situations. The complexity of the algorithm of image processing for the human detection is an exponential one and in the future we hope to simplify it.

References

[1] *Directive 2002/22/EC of the European Parliament and of the Council of 7 March 2002 on universal service and users' rights relating to electronic communications networks and services (Universal Service Directive)* http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32002L0022:EN:HTML:, last visited on 29 March 2009

[2] *Products: Robotino*, http://www.festo-didactic.com/int-en/learning-systems/new-robotino/, last visited 29 March 2008

[3] *National Institute for Urban Search and Rescue: Teams, Techniques, Technology & Training* http://niusr.org/ , last visited 28 March 2009

[4] *Human Detection by Searching in 3D Space Using Camera and Scene Knowledge* http://figment.cse.usf.edu/ sfefilat/data/papers/WeAT2.5.pdf

[1] Faculty of Mathematics and Computer Science, Babes-Bolyai University, Kogalniceanu 1, Cluj-Napoca, 400084, Romania

E-mail address: alina.miron@hotmail.com

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 171–174

APPLICATIONS OF SELF-ORGANIZING MAPS IN BIO-INSPIRED ARTIFICIAL VISION MODELS

PAUL V. BORZA[(1)], OANA GUI[(2)], AND DAN DUMITRESCU[(3)]

Abstract. This paper describes the need for bio-inspired vision models when detecting dissimilarities and rejecting similarities in consecutively taken photos in various environments over a certain period of time. Such an approach to this problem are self-organizing maps that are briefly discussed, focusing more on the algorithmic process that was used to train these. This machine learning technique was applied in this research regarding finding regions of interest with the goal of accomodating the varios properties like weather change, luminance change, camera shake etc. that invariably occur, and a naive approach would fail.

1. Introduction

The paper aims at drawing attention to the depth and breadth of biologically inspired design as practiced by scientists and engineers. Both a method and goal, the biologically inspired design expands across many disciplines that currently are organized around functional criteria, or around levels of inquiry. Two aspects need to be taken into consideration: first, the need to recognize the fields, problems and applications for which biological inspiration can have an impact and second, understanding the basic steps required for a successful merge of biological and engineering knowledge. The main concern is to transpose biological principles to engineering designs and applications.

Biologically inspired design is usually thought of as being problem based, that is, motivated by the need to generate an improved solution to a particular technical challenge. An opposed approach is to consider "biological solutions" as a starting point and seek out particular technical solutions for which the system is appropriate. This solution-based approaches may encourage the application of biological principles in ways that would not be immediately obvious when viewed from a problem-based perspective.

The ability to examine the implementation of particular principles across many organisms allows us to discover both the generality and robustness of a particular "biological solution", the Region of Interest use, in our case. Common in most biologica organisms, this method significantly reduces the amount of information passed

2000 *Mathematics Subject Classification.* 68U10.
Key words and phrases. self-organizing maps, bio-inspired, artificial, vision models.

to subsequent processing layers. A second advantage of the opportunity for comparison is that biological systems often are dedicated to solve a specific problem within the more general problem space. In our research we have applied a machine learning technique (SOM) for detecting and tracking differences in consecutively taken images.

2. Motivation

Given a temporal sequence of VGA snapshots at 30 FPS (frames per second), the described algorithm detects differences and rejects similarities from these consecutive photos. The main directions in developing a SOM-based solution were accuracy and speed of the algorithm in order to run in real-time as the flow of images becomes available.

Self-organizing maps are used to produce a similarity (and in the same time a dissimilarity) graph of the input data. These convert the nonlinear statistical relationships between high-dimensional data into simple geometric relationships of their image points on a regular two-dimensional grid of nodes [1].

3. The algorithm

The SOM defines a mapping from the input data space $\mathfrak{R}^n$ onto a two-dimensional array of nodes. For every node in the array, there is an associated *model vector*, a *reference vector*, that is defined as $m_i = [\mu_{i1}, \mu_{i2}, \ldots, \mu_{in}]^T \in \mathfrak{R}^n$. The input domain is represented by 8-bit color depth pixels that are characterized by each of the $R, G, andB$ channels, making $n = 3$; each channel spans on the $[0 \ldots 255]$ interval.

All input vectors were considered to be connected to all neurons in parallel via the reference vectors μ_{ij}. The comparison of the input vector with all m_i was done using the *Euclidean distance*, such that the location of the *best-matching unit* could rapidly be found without much processing power [2].

For each step, each *frame*, the following recursive formula was used to train the neurons to the new environment, which usually differs from one frame to the other in terms of *angle* (e.g. camera shake), *luminance* (e.g. change of weather), and other factors that disturb the scene

$$m_i(t+1) = m_i(t) + h_{ci}(t)[x(t) - m_i(t)] \ . \tag{1}$$

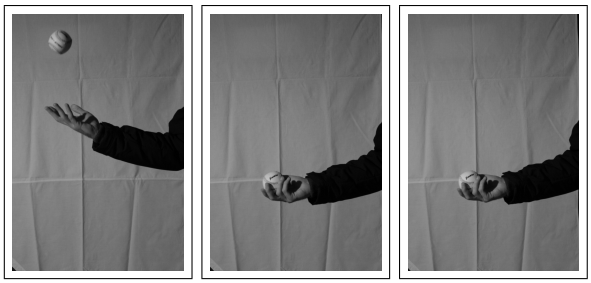
It is necessary to have the *neighborhood function* $h_{ci}(t) \to 0$ when $t \to \infty$. This approach considers a *neighborhood set*, N_c, of nodes around the *best-matching unit* c, and defines the function as

$$h_{ci}(t) = \begin{cases} \alpha(t), & \text{if } i \in N_c \\ 0, & \text{if } i \notin N_c \end{cases} \ . \tag{2}$$

The *learning-rate factor* was considered to be bounded by $0 < \alpha(t) < 1$ and after several iterations, together with $N_c(t)$, to monotonically decrease in time. For the first iteration, the neighborhood set was taken as large as the whole diameter of the map, while the learning-rate factor almost as high as its upper limit.

4. Test bench

Three photos taken with a Canon EOS 40D in less than half of second at time t, $t+1$ and time $t+2$ were considered.



5. Rejecting similarities and detecting dissimilarities

Each pair of the form t and $t+1$ was taken for comparison. In order to decide whether these photos have changed, an *average feature vector* (i.e. a hash) measuring the intensities of the colors was calculated for each photo p_t and p_{t+1} taken at time t, respectively $t+1$. If the difference of the norm of the average feature vectors for p_t and p_{t+1} was less than ϵ, then the photos were said to be the same and no further computation was necessary. However, if the previous condition failed, then using a *divide et impera* approach, each of the p_t and p_{t+1} photos were divided into four smaller ones, and the whole algorithm ran on each of them as if it would work on completely new photos. Once the algorithm reached a certain depth, it stopped and compared each pixel individually from one photo to the other.

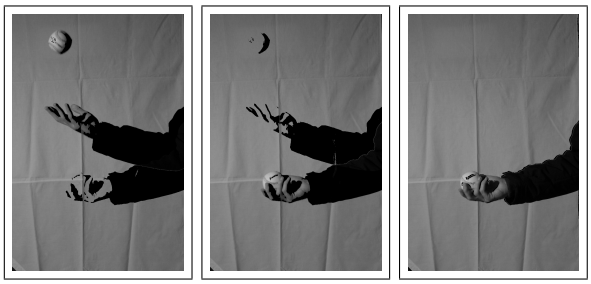
The decision whether *two pixels* located at the same position in each photo pair *are similar or dissimilar* was given by the *distance* between the location of the best-matching unit $BMU_{t,(x,y)}$ (i.e. the pixel located at (x,y) in the first photo) and the location of the other $BMU_{t+1,(x,y)}$ (i.e. the pixel located at (x,y) in the second photo). If the distance between these best matching units exceeded a certain threshold, then the pixels were said to be dissimilar (i.e. changed from time t to $t+1$), otherwise the pixels were said to be similar.

6. Results

Self-organizing maps, an unsupervised machine learning technique, were able to train themselves using the above formulas and approaches; at each iteration, the grid of neurons becomes more and more efficient in detecting differences and is able to rebuild itself regarding the contents of the photos and its properties.

Detecting differences in consecutively taken photos not only requires very efficient algorithm implementations, but also the ability to distinguish between translations, rotations and other shifts that would not be detected by naive approaches. The above

trained self-organizing map was able to see only the *correct differences* between the photos used in the test bench.



7. Conclusions

This paper shows that self-organizing maps have the ability to overcome usual problems such as change of luminance, camera shakes etc. that invariably occur when a photographer takes photos, or when a camera records video. Being able to detect *regions of interest* allows us to focus only on that particular area, recognize the moving object and even estimate its future trajectory.

References

[1] Teuvo Kohonen. *Self-organizing maps*. Springer, third edition, 2001.
[2] Sergios Theodoridis and Konstantinos Koutroumbas. *Pattern recognition*. Academic Press, first edition, 1999.

[1] Faculty of Mathematics and Computer Science, Babeş-Bolyai University, Mihail Kogalniceanu 1, 400084 Cluj-Napoca, Romania
E-mail address: paul@borza.ro

[2] University of Arts and Design of Cluj-Napoca
E-mail address: oana.gui@gmail.com

[3] Faculty of Mathematics and Computer Science, Babeş-Bolyai University, Mihail Kogalniceanu 1, 400084 Cluj-Napoca, Romania
E-mail address: ddumitr@cs.ubbcluj.ro

KNOWLEDGE ENGINEERING: PRINCIPLES AND TECHNIQUES
Proceedings of the International Conference on Knowledge Engineering,
Principles and Techniques, KEPT2009
Cluj-Napoca (Romania), July 2–4, 2009, pp. 175–178

Q-LEARNING AND POLICY GRADIENT METHODS

HUNOR JAKAB AND LEHEL CSATÓ

Abstract. Many real-world tasks require a robotic agent to adapt its behavior to certain environmental conditions and to acquire knowledge without user interaction. In reinforcement learning knowledge is usually acquired without preexisting training data, thereby making the learning process more "natural". In this paper we investigate two reinforcement learning methods and present a *simulation environment* where we test their performance. The simulation environment allows the testing of various reinforcement algorithms without a need for the physical robot. Its advantage is that it can be used to perform benchmarks and evaluations of different learning algorithms.

1. Introduction

The development of efficient learning methods is an essential field of robotic research. Several real-world tasks require a robotic agent to adapt its behaviour to new environmental conditions – usually unexpected ones – and to acquire knowledge within these changed conditions. This should be done without user intervention, and this framework is usually called reinforcement learning. These problems can efficiently be solved using statistical methods linked with dynamic programming, based on the estimation of *rewards* linked to action-state pairs. Decision making, in situations where delayed reinforcement is essential, can be modelled with Markov Decision Processes (MDP), or with Partially Observable MDPs [Dar and Mansour, 2003]. In the following we give a definition of MDPs and the problem statement in an informative way, then we present two learning algorithms for reinforcement learning. The comparative behaviour of the two methods is briefly discussed in Section 5, where the comparisons use the simulation environment developed by us.

2. Markov Decision Processes

A Markov decision process [Kaelbling et al., 1996] is a quadruple $M\,(S, A, P_a, R_a)$ with the following components: (1) S is the set of states; (2) A the set of actions; (3) $P_a : S \times A \to p(S)$, $a \in A$ such that $P_a(x_1, x_2) \stackrel{\text{def}}{=} p(x_2|x_1, a)$ is the conditional probability of a transition from state x_1 to x_2 when executing action a; (4) $R_a : S \to \mathbb{R}$, $a \in A$, $R_a(x)$ denoting the immediate reward received after executing action a from state x. MDPs provide the framework to solving the reinforcement learning: the goal is to find a policy π, a probability distribution over actions for a given state x, denoted $\pi(a|x)$ and valid for all states $x \in S$. The policy maximizes the expected

2000 *Mathematics Subject Classification.* 68T05, 68T40, 60J25.
Key words and phrases. reinforcement learning, Markov decision processes, robotics.

cumulative reward, *i.e.* the discounted infinite horizon summation:

$$J_\pi = E\left[\sum_{t=0}^{\infty} \gamma^t R^\pi_{a_t}(x_t)\right] \tag{1}$$

where $E[\cdot]$ is the expected value over the policy, $0 < \gamma < 1$ is the discount rate, and $R^\pi_a(x)$ is the reward *conditioned* on the policy π. The expression on the right hand side of (1) is analytically tractable and the existence of a solution has been proven [Kaelbling et al., 1996]. However, finding the solution is often difficult and there are two possible directions. A first solution is to reconstruct the MDP and *afterwards* find an optimal policy for the reconstructed model. The second approach is based on iterative methods that perform a stepwise update of information, based on which they estimate the optimal policy [Dar and Mansour, 2003].

3. Q-Learning

The Q-learning algorithm [Watkins, 1989] is a direct minimization approach and it performs *value iteration* [Bertsekas, 1995] to approximate the policy that maximizes cumulative reward defined in (1). The *value of a state* can be expressed as $V(x) = E\left[\sum_{t=1}^{\infty} \gamma^t R_{a_t}(x_t)\right]$, the same as in (1) except that the average is with respect to the *optimal policy*. According to the Bellman optimality equation [Bertsekas, 1995], the optimal state value has the following form:

$$V^*(x) = \max_{a \in A}\left(R_a(x) + \gamma \sum_{z \in S} V^*(x) P_a(x, z)\right) \tag{2}$$

Finding the equilibrium state using (2) is usually difficult because one has to integrate over *all* successor states which is computationally infeasible. To avoid this, the Q-learning algorithm uses a value function on state-action space, instead of state-space alone: $Q : S \times A \to \mathbb{R}$. The value of a state alone is the maximum Q-value among all *actions*. The update rules are found by reformulating the optimality condition in (2) and applying prediction difference to adjust the estimate of $Q(x, a)$:

$$\begin{aligned} Q^*(x_t, a_t) \quad &= \quad Q^*(x_t, a_t) \\ &\quad -\alpha\left(R_a(x_t) + \gamma \max_{a_{t+1}} Q^*(x_{t+1}, a_{t+1}) - Q^*(x_t, a_t)\right) \end{aligned} \tag{3}$$

where $0 < \alpha < 1$ is the learning rate, and $Q^*(x, a)$ is the estimation of the optimal Q-value function. By applying equation (3) infinitely often to all the state-action pairs, $Q^*(x, a)$ converges to its optimal value [Sutton and Barto, 1998].

4. Policy Gradient methods

Policy gradient (PG) algorithms optimize the *parameters* $\boldsymbol{\theta}$ of a parametric policy, where the optimization is being done with respect to the expected reward $J(\boldsymbol{\theta})$. We thus need a good approximation to the policy gradient, as in equation (4) with respect to the episodic reward $R(\tau) = \sum_{t=0}^{H-1} \gamma^t R_{a_t}(x_t)$, and the update is done based on the expression in (5) [Peters et al., 2007], where α is a learning rate, i is the current

update step, and τ stands for a history of controller outputs for an episode of length H:

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \int \nabla_{\boldsymbol{\theta}} p_{\boldsymbol{\theta}}(\tau) R(\tau) d_\tau \tag{4}$$

$$\boldsymbol{\theta}_{i+1} = \boldsymbol{\theta}_i + \alpha_i \nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \tag{5}$$

Likelihood ratio methods [Rückstieß et al., 2008] provide a good way of accomplishing this. In order to incorporate exploratory behavior in the policy we perturb the outputs of the controller by random Gaussian noise

$$\boldsymbol{a} = f(\boldsymbol{x}, \boldsymbol{\theta}) + \boldsymbol{e} \quad \text{where} \quad \boldsymbol{e} \sim \mathrm{N}(0, \sigma^2)$$

Knowing that $p_{\boldsymbol{\theta}}(\tau) = p(x_0) \prod_{t=0}^{H-1} \pi(a_t|x_t) p(x_{t+1}|x_t, a_t)$ and following the derivation from [Wierstra et al., 2007] the gradient is expressed as:

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = E\left(\sum_{t=0}^{H-1} \nabla_{\boldsymbol{\theta}} \log \pi(a_t|x_t) R(\tau) \right)$$

This is approximated with Markov sampling by taking the average over a number of controlled output histories [Williams, 1992]. It was shown in [Sutton et al., 1999] that PG algorithms converge to a local optimum.

5. Comparative behavior

To analyse the behavior of the described algorithms we created a virtual robot with two degrees of freedom, as in Fig. 1.a, and we simulated it's dynamics using the ODE physics engine.[1] Both the Q-learning and the policy gradient algorithms were implemented, the task being the *movement* in a certain direction. There are several parameters that affect the Q-learning, we plotted the convergence rates and the efficiency of the algorithm in Fig. 1.b. For Q-learning there is no need for a parametric model, but it leads us to the drawback that one has to discretize the state-space as seen if Fig. 1.c, limiting its applicability to problems with low dimensions. The policy gradient algorithm on the other hand is capable of handling continuous search spaces with the help of function approximators(in our experiments we used neural networks for f with variable complexity), but gradient variance makes the algorithm converge more slowly. Computational costs of the PG algorithm are also much higher, as a result of the complexity of gradient evaluation. In both algorithms the hand-tuning of parameters such as α, γ, σ^2 is crucial for good performance.This work was supported by grant POSDRU/6/1.5/S/3/2008-ID 5216 and PNII 11-039/2007.

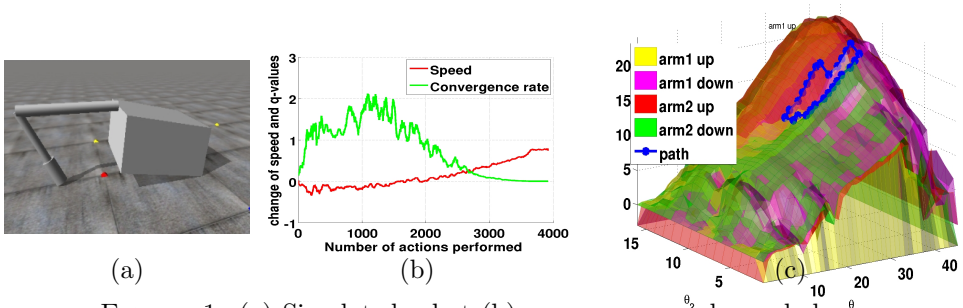
References

D. Bertsekas. *Dynamic Programming and Optimal Control.* Athena Scientific, Belmont, MA, 1995.

E. E. Dar and Y. Mansour. Learning rates for q-learning. *Journal of Machine Learning Research*, 5:1–25, 2003.

L. P. Kaelbling, M. L. Littman, and A. W. Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.

[1]Open Dynamics Engine: http://www.ode.org.



(a) (b) (c)

FIGURE 1. (a) Simulated robot (b) convergence and speed changes (c) Q-value function and trajectory

J. Peters, E. Theodorou, and S. Schaal. Policy gradient methods for machine learning, 2007.

T. Rückstieß, M. Felder, and J. Schmidhuber. State-dependent exploration for policy gradient methods. In *ECML/PKDD*, pages 234–249, 2008.

R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

R. S. Sutton, D. A. McAllester, S. P. Singh, and Y. Mansour. Policy gradient methods for reinforcement learning with function approximation. In S. A. Solla, T. K. Leen, and K.-R. Müller, editors, *NIPS*, pages 1057–1063, 1999.

C. J. C. H. Watkins. *Learning from Delayed Rewards*. PhD thesis, King's College, London, 1989.

D. Wierstra, A. Foerster, J. Peters, and J. Schmidthuber. Solving deep memory POMDPs with recurrent policy gradients, 2007.

R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256, 1992.

DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE, BABEŞ-BOLYAI UNIVERSITY, 1 KOGALNICEANU STR., 400084 CLUJ-NAPOCA, ROMANIA

E-mail address: {jakabh,csatol}@cs.ubbcluj.ro