Applied Mathematical Sciences, Vol. 6, 2012, no. 14, 689 - 696

On the Problem of Placement of Visual Landmarks

Anna Gorbenko

Department of Intelligent Systems and Robotics
Ural Federal University
620083 Ekaterinburg, Russia
gorbenko.ann@gmail.com

Vladimir Popov

Department of Intelligent Systems and Robotics
Ural Federal University
620083 Ekaterinburg, Russia
Vladimir.Popov@usu.ru

Abstract

In recent years among developers of robotic software formed a direction of the development of individual solvers. Such solvers are designed for specific hard problems. In this paper we consider an approach to design of an efficient solver for the problem of placement of visual landmarks.

Mathematics Subject Classification: 68T40

Keywords: mobile robots, visual landmarks, genetic algorithms

1 Introduction

Many robotic problems are computationally hard (e.g. [1] – [7]). In particular, we can mention planning problems, pattern recognition, pattern matching, localization problems, mapping problems, SLAM (simultaneous localization and mapping) and many others. Implementation of various heuristics for solving such problems greatly complicates the development of efficient software for robotic systems. In recent years among developers of robotic software formed a direction of the development of individual solvers. Such solvers are designed for specific hard problems [8] – [11].

Using the selection, extraction and recognition of visual landmarks has been extensively applied for mobile robot navigation (see e.g. [12] – [15]). Visual landmarks robot navigation approaches select certain features in the snapshot image as landmarks, and try to establish correspondences between these landmarks and features extracted from the current view image. Such approaches differ with respect to the strategy for selecting the landmarks. Some methods strive to extract maximally distinctive features (see e.g. [16] – [18]). Other approaches use less unique features such as dark or bright sectors (see e.g. [19] – [22]), Harris corners (see e.g. [23], [24]), or coloured regions (see e.g. [25], [26]).

In [3] considered the problem of placement of visual landmarks (VL). In particular, in [3] proved that VL is **NP**-complete. Also in [3] considered an approach to design of an efficient solver for VL. This approach is based on constructing an explicit reduction to the problem MAXSAT. In this paper we continue to develop the approach. In particular, we consider an explicit reduction from VL to 3SAT and propose a new genetic algorithm.

2 Problem definition

We consider a problem of placement of visual landmarks in the discrete space Z^2, where Z is the set of integers. Also we identify every element x of Z^2 with a square with sides equal to one and the center in x. A set of points in Z^2 which of interest to navigation we denote by N. Note that N is not necessarily a connected area. For instance, we can be interested only in surface facilities and a part of the surrounding area can be covered by water. Let S be a set of points in Z^2 which permissible to placement of landmarks. It is natural to assume that we are dealing with some limited region R such that $N \subseteq R$, $S \subseteq R$. Since the deployment region R can contain obstacles or visual landmarks can be visible not from all points of space, it is natural to assign for each point of the set S its own field of vision which is defined by the function $F : S \to 2^R$. We can suppose that F is given by the sequence of pairs consisting of elements of S and corresponding subsets. We also consider some constant d which determines a minimal number of necessary landmarks. Note that the value of d, usually, does not exceed $4 - 10$ (see e.g. [27]). In the form of a satisfiability problem VL can be formulated as follows:

THE PROBLEM OF PLACEMENT OF VISUAL LANDMARKS (VL):

INSTANCE: *A finite set R, $S \subseteq R$, $N \subseteq R$, $F : S \to 2^R$, a positive integer parameter k, and a positive integer constant d.*

QUESTION: *Is there $T \subseteq S$ such that $|T| \leq k$ and for any $y \in N$ there is $D \subseteq T$ such that $|D| \geq d$ and $y \in F(x)$ for all $x \in D$?*

3 A logical model for VL

Consider the satisfiability problem (SAT): *given a boolean expression in conjunctive normal form (CNF), is it satisfiable?* SAT was the first known **NP**-complete problem. Considered also different variants of SAT. For example, 3SAT: *given a boolean expression in CNF with 3 variables per clause (3-CNF), is it satisfiable?*

Let $S = \{a_1, a_2, \ldots, a_n\}, N = \{b_1, b_2, \ldots, b_m\}$. For all i, $1 \le i \le m$, consider a set $M_i = \{p \mid b_i \in F(a_p)\}$. It is easy to see that the system of sets M_i, $1 \le i \le m$, can be constructed in polynomial time. Obviously, if we have an inequality $|M_i| < d$ for some i, then the solution of VL is negative. Thus, in further, without loss of generality, we can assume that $|M_i| \ge d$ for all i. Let

$$\begin{aligned}
\varphi_1 &= \wedge_{1\le i\le k}((\vee_{1\le j\le n} x[i,j]),\\
\varphi_2 &= \wedge_{1\le i\le k, 1\le j[1]<j[2]\le n}(\neg x[i,j[1]] \vee \neg x[i,j[2]]),\\
\varphi_3 &= \wedge_{1\le i[1]<i[2]\le k, 1\le j\le n}(\neg x[i[1],j] \vee \neg x[i[2],j]),\\
\varphi &= \varphi_1 \wedge \varphi_2 \wedge \varphi_3,\\
\psi_1 &= \wedge_{1\le i\le m, 1\le j\le d,}((\vee_{1\le l\le k} y[i,j,l]),\\
\psi_2 &= \wedge_{1\le i\le m, 1\le j\le d, 1\le l[1]<l[2]\le k}(\neg y[i,j,l[1]] \vee \neg y[i,j,l[2]]),\\
\psi_3 &= \wedge_{1\le i\le m, 1\le j[1]<j[2]\le d, 1\le l\le k}(\neg y[i,j[1],l] \vee \neg y[i,j[2],l]),\\
\psi &= \psi_1 \wedge \psi_2 \wedge \psi_3,\\
\rho &= \wedge_{1\le i\le m, 1\le j\le d, 1\le l\le k} \neg y[i,j,l] \vee (\vee_{j\in M_i} x[l,j]).
\end{aligned}$$

Theorem 3.1 *Given a finite set R, $S \subseteq R$, $N \subseteq R$, $F : S \to 2^R$, a positive integer parameter k, and a positive integer constant d. There is $T \subseteq S$ such that $|T| \le k$ and for any $y \in N$ there is $D \subseteq T$ such that $|D| \ge d$ and $y \in F(x)$ for all $x \in D$ if and only if $\xi = \varphi \wedge \psi \wedge \rho$ is satisfiable.*

PROOF. Given a finite set R, $S \subseteq R$, $N \subseteq R$, $F : S \to 2^R$, a positive integer parameter k, and a positive integer constant d.

Suppose that there is $T \subseteq S$ such that $|T| \le k$ and for any $y \in N$ there is $D_y \subseteq T$ such that $|D_y| \ge d$ and $y \in F(x)$ for all $x \in D_y$. Without loss of generality we can assume that $|T| = k$ and $|D_y| = d$. Let $T = \{a_{i_1}, a_{i_2}, \ldots, a_{i_k}\}$, $D_y = \{a_{i_{l[1]}}, a_{i_{l[2]}}, \ldots, a_{i_{l[d]}}\}$. Suppose that $x[l,j] = 1$ if and only if $i_l = j$ where $1 \le l \le k$, $1 \le j \le n$. Let $y[s,j,t] = 1$ if and only if $a_{i_{l[j]}} \in D_{b_s}$ and $l[j] = t$ where $1 \le s \le m$, $1 \le j \le d$, $1 \le t \le k$. It is easy to check that in this case $\xi = 1$.

Now suppose that $\xi = 1$. It it clear that in this case $\varphi = 1$. In view of $\varphi_1 = 1$, for all i, $x[i,j] = 1$ at least for one value of j. Since $\varphi_2 = 1$, it is easy to see that, for all i, there is only one value of j such that $x[i,j] = 1$. In view of $\varphi_3 = 1$, it is clear that if $x[i[1], j] = x[i[2], j] = 1$, then $i[1] = i[2]$. Therefore, we can consider values of $x[i, j]$ as a choice of elements of S. In particular, we can suppose that $x[s, t] = 1$ if and only if $a_{i_s} \in T$ and $i_s = t$. Since if $x[i[1], j] = x[i[2], j] = 1$, then $i[1] = i[2]$, it is clear that $|T| = k$. Similarly, we can consider values of $y[i, j, l]$ as a choice of elements of S. In particular, we can suppose that $y[s, j, t] = 1$ if and only if $a_{i_{l[j]}} \in D_{b_s}$ and $l[j] = t$ where $1 \le s \le m$, $1 \le j \le d$, $1 \le t \le k$. Since $\rho = 1$, it is easy to check that There is $T \subseteq S$ such that $|T| \le k$ and for any $y \in N$ there is $D \subseteq T$ such that $|D| \ge d$ and $y \in F(x)$ for all $x \in D$. $\square$

Clearly, ξ is a CNF. So, ξ give us an explicit reduction from VL to SAT. Note that

$$\begin{aligned}
\alpha &\Leftrightarrow (\alpha \vee \beta_1 \vee \beta_2) \wedge (\alpha \vee \neg\beta_1 \vee \beta_2) \wedge \\
&\quad\ (\alpha \vee \beta_1 \vee \neg\beta_2) \wedge (\alpha \vee \neg\beta_1 \vee \neg\beta_2), && (1)\\
\vee_{j=1}^{l}\alpha_j &\Leftrightarrow (\alpha_1 \vee \alpha_2 \vee \beta_1) \wedge (\wedge_{i=1}^{l-4}(\neg\beta_i \vee \alpha_{i+2} \vee \beta_{i+1})) \wedge \\
&\quad\ (\neg\beta_{l-3} \vee \alpha_{l-1} \vee \alpha_l), && (2)\\
\alpha_1 \vee \alpha_2 &\Leftrightarrow (\alpha_1 \vee \alpha_2 \vee \beta) \wedge (\alpha_1 \vee \alpha_2 \vee \neg\beta), && (3)\\
\vee_{j=1}^{4}\alpha_j &\Leftrightarrow (\alpha_1 \vee \alpha_2 \vee \beta_1) \wedge (\neg\beta_1 \vee \alpha_3 \vee \alpha_4) && (4)
\end{aligned}$$

where $l > 4$. Using relations (1) – (4) we can obtain an explicit transformation ξ in τ such that $\xi \Leftrightarrow \tau$ and τ is a 3-CNF. Clearly, τ give us an explicit reduction from VL to 3SAT.

4 Experimental Results

In previous section we obtain explicit reductions from VL to SAT and 3SAT. We use algorithms fgrasp and posit from [28]. Also we design our own genetic algorithm for SAT which based on algorithms from [28].

Consider a boolean function $g(x_1, x_2, \ldots, x_n) = \wedge_{i=1}^{m}\mathcal{C}_i$, where $m \ge 1$, and each of the $\mathcal{C}_i$ is the disjunction of one or more literals. Let $|\mathcal{C}_i|$ be a number of literals in $\mathcal{C}_i$. Let $occ(x_i, g)$ be a number of occurrences of x_i in g. Respectively, let $occ(\neg x_i, g)$ be a number of occurrences of x_i in g. For example, if $g = (x_1 \vee x_2) \wedge (\neg x_2 \vee x_3) \wedge (x_1 \vee x_4) \wedge (\neg x_1 \vee x_5)$, then $occ(x_1, g) = 2$, $occ(\neg x_1, g) = 1$.

We consider a number of natural principles that define importance of a variable x_i for satisfiability of boolean function g. These principles suggest us correct values of variables.

1. If $occ(x_i, g) \ge 0$ and $occ(\neg x_i, g) = 0$, then $x_i = 1$.
2. If $occ(x_i, g) = 0$ and $occ(\neg x_i, g) \ge 0$, then $x_i = 0$.
3. If $occ(x_i, g) > occ(\neg x_i, g)$, then $x_i = 1$.
4. If $occ(x_i, g) < occ(\neg x_i, g)$, then $x_i = 1$.
5. If $x_i = \mathcal{C}_j$ for some j, then $x_i = 1$.

6. If $\neg x_i = \mathcal{C}_j$ for some j, then $x_i = 0$.
7. If $\min_{occ(x_i,\mathcal{C}_j)>0} |\mathcal{C}_j| \leq \min_{occ(\neg x_i,\mathcal{C}_j)>0} |\mathcal{C}_j|$, then $x_i = 1$.
8. If $\min_{occ(x_i,\mathcal{C}_j)>0} |\mathcal{C}_j| \geq \min_{occ(\neg x_i,\mathcal{C}_j)>0} |\mathcal{C}_j|$, then $x_i = 0$.

Based on these principles, we can consider the following eight types of commands: $P_1, P_2, \ldots, P_8$. Also we consider the following three commands for run algorithms: Try_fgrasp, Try_posit, and Try_ga, where Try_ga runs a simple genetic algorithm. Denote by $\mathcal{R}$ the set of commands of these eleven types. Arbitrary element of $\mathcal{R}^*$ it is possible to consider as a program for finding values of variables of a boolean function. We assume that such programs are executed on a cluster. Execution of each of commands of type P_i reduces the number of variables of a boolean function by one. Execution of each of commands Try_fgrasp, Try_posit, and Try_ga consists in the run of corresponding algorithm for current boolean function on a separate set of calculation nodes and the transition to the next command. Algorithms fgrasp and posit we run only on one calculation node. Genetic algorithms can be used in parallel execution. We use auxiliary genetic algorithm which determine the number of calculation nodes.

Initially, we selected a random subset of $\mathcal{R}^*$. We use a genetic algorithm to select a program from the current subset of $\mathcal{R}^*$ and a genetic algorithm for evolving the current subset of $\mathcal{R}^*$. The evolution of the current subset of $\mathcal{R}^*$ implemented on a separate set of calculation nodes. For every subsequent boolean functions it is used the current subset of $\mathcal{R}^*$ which is obtained by taking into account the results of previous runs.

We use heterogeneous cluster based on three clusters (Cluster USU, Linux, 8 calculation nodes, Intel Pentium IV 2.40GHz processors; umt, Linux, 256 calculation nodes, Xeon 3.00GHz processors; um64, Linux, 124 calculation nodes, AMD Opteron 2.6GHz bi-processors) [29].

We use a generator of natural instances for VL. Algorithms fgrasp and posit used only for 3SAT. For SAT used simple genetic algorithm (SGA), and our algorithm (OA). Selected experimental results are given in Tables 1, 2.

Table 1: Experimental results for 3SAT.

time	fgrasp	posit	SGA	OA
average	13.5 min	12.8 min	11.1 min	8.8 min
maximum	12.1 h	11.8 h	12.6 h	11.6 h
best	37 sec	31 sec	19 sec	11 sec

Table 2: Experimental results for MAXSAT [3], SAT and parallel run of MAXSAT, SAT and 3SAT.

time	SGA (MAXSAT)	SGA (SAT)	OA (SAT)	parallel run
average	10.9 min	11.4 min	9.1 min	2.9 min
maximum	12.4 h	12.5 h	11.4 h	3.5 h
best	26 sec	30 sec	15 sec	17 sec

5 Conclusion

In this paper we continue to develop the approach of [3]. In particular, we consider an explicit reduction from VL to 3SAT. Although we assume that d is a constant, reduction to 3SAT can be used for arbitrarily large values of d (in contrast to the reduction to MAXSAT considered in [3]). Also we propose a new genetic algorithm. In our experiments the best performance was achieved when using the parallel running of algorithms.

ACKNOWLEDGEMENTS. The work was partially supported by Analytical Departmental Program "Developing the scientific potential of high school" 2.1.1/14055.

References

[1] R. Borie, C. Tovey, and S. Koenig, Algorithms and Complexity Results for Pursuit-Evasion Problems, *Proceedings of the International Joint Conference on Artificial Intelligence*, (2009), 59-66.

[2] A. Gorbenko, A. Lutov, M. Mornev, and V. Popov, Algebras of Stepping Motor Programs, *Applied Mathematical Sciences*, 5 (2011), 1679-1692.

[3] A. Gorbenko, M. Mornev, and V. Popov, The problem of placement of visual landmarks, *SYRCoSE 2011*, (2011), 134-141.

[4] A. Gorbenko, M. Mornev, and V. Popov, Planning a Typical Working Day for Indoor Service Robots, *IAENG International Journal of Computer Science*, 38 (2011), 176-182.

[5] A. Gorbenko, V. Popov, and A. Sheka, Robot Self-Awareness: Temporal Relation Based Data Mining, *Engineering Letters*, 19 (2011), 169-178.

[6] P. Sala, R. Sim, A. Shokoufandeh, and S. Dickinson, Landmark selection for vision-based navigation, *IEEE Transactions on Robotics*, 22 (2006), 334-349.

[7] X. Zheng, S. Koenig, D. Kempe, and S. Jain, Multi-Robot Forest Coverage for Weighted and Unweighted Terrain, *IEEE Transactions on Robotics*, 26 (2010), 1018-1031.

[8] D. Christensen, E. Ostergaard, and H. Lund, Metamodule control for the atron self-reconfigurable robotic system, *Proceedings of the 8th International Conference on Intelligent Autonomous Systems*, (2004), 685-692.

[9] D. Dewey, S. Srinivasa, M. Ashley-Rollman, M. De Rosa, P. Pillai, T. Mowry, J. Campbell, and S. Goldstein, Generalizing Metamodules to Simplify Planning in Modular Robotic Systems, *Proceedings of IEEE/RSJ 2008 International Conference on Intelligent Robots and Systems*, (2008), 1338-1345.

[10] D. Rus and M. Vona, Crystalline Robots: Self-reconfiguration with Unit-compressible Modules, *Autonomous Robots*, 10 (2001), 107-124.

[11] C. Ünsal and P. Khosla, A Multi-layered Planner for Self-Reconfiguration of a Uniform Group of I-Cube Modules, *Proceedings of IEEE/RSJ 2001 International Conference on Intelligent Robots and Systems*, 1 (2001), 598-605.

[12] M. Elmogy, Landmark manipulation system for mobile robot navigation, *2010 International Conference on Computer Engineering and Systems*, (2010), 120-125.

[13] J. Hayet, F. Lerasle, and M. Devy, A visual landmark framework for indoor mobile robot navigation, *IEEE International Conference on Robotics and Automation*, 4 (2002), 3942-3947.

[14] J. Hayet, F. Lerasle, and M. Devy, A visual landmark framework for mobile robot navigation, *Image and Vision Computing*, 25 (2007), 1341-1351.

[15] D. Meyer-Delius, M. Beinhofer, A. Kleiner, and W. Burgard, Using Artificial Landmarks to Reduce the Ambiguity in the Environment of a Mobile Robot, *Proceedings of the IEEE International Conference on Robotics and Automation*, (2011), 270-275.

[16] J. Hong, X. Tan, B. Pinette, R. Weiss, and E. Riseman, Image-based homing, *IEEE Control Syst.*, 12 (1992), 38-45.

[17] D. Lowe, Distinctive image features from scale-invariant keypoints, *Int. J. Comput. Vision*, 60 (2004), 91-110.

[18] S. Se, D. Lowe, and J. Little, Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks, *Int. J. Robot. Res.*, 21 (2001), 735-758.

[19] B. Cartwright and T. Collett, Landmark learning in bees, *J. Comp. Physiol. A*, 151 (1983), 521-543.

[20] B. Cartwright and T. Collett, Landmark maps for honeybees, *Biol. Cybernet.*, 57 (1987), 85-93.

[21] D. Lambrinos, R. Möller, T. Labhart, R. Pfeifer, and R. Wehner, A mobile robot employing insect strategies for navigation, *Robot. Auton. Syst.*, 30 (2000), 39-64.

[22] K. Weber, S. Venkatesh, and M. Srinivasan, Insect-inspired robotic homing, *Adaptive Behav.*, 7 (1999), 65-97.

[23] A. Vardy and F. Oppacher, Low-level visual homing, *Proceedings of the 7th European Conference on Artificial Life*, (2003), 875-884.

[24] A. Vardy and F. Oppacher, Anatomy and physiology of an artificial vision matrix, *Biologically Inspired Approaches to Advanced Information Technology: First International Workshop*, (2004), 290-305.

[25] T. Goedemé, M. Nuttin, T. Tuytelaars, and L. Van Gool, Vision-based intelligent wheel chair control: rhe role of vision and inertial sensing in topological navigation, *J. Robot. Syst.*, 21 (2004), 85-94.

[26] S. Gourichon, J.-A. Meyer, S. Ieng, L. Smadja, and R. Benosman, Estimating ego-motion using a panoramic sensor: comparison between a bio-inspired and a camera-calibrated method, *AISB03 Symposium on Biologically Inspired Vision, Theory and Application*, (2003), 91-101.

[27] R. Lerner, E. Rivlin, and I. Shimshoni, Landmark Selection for Task-Oriented Navigation, *IEEE Transactions on Robotics*, 23 (2007), 494-505.

[28] http://people.cs.ubc.ca/∼hoos/SATLIB/index-ubc.html

[29] http://parallel.imm.uran.ru/mvc_now/hardware/supercomp.htm

Received: September, 2011