Pursuit Games in Obstacle Strewn Fields Using Distance Transforms

Ray Jarvis and Mohamed Marzouqi
Intelligent Robotics Research Centre, Monash University
Australian Research Council Centre for Perceptive and
Intelligent Machines in Complex Environments
{ray.jarvis, mohamed.marzouqi}@eng.monash.edu.au

Abstract

In pursuit games it is of interest to ascertain whether, in various obstacle strewn spaces, the predator can if at all possible, intercept the prey, which in turn tries to reach a safe haven before being intercepted. Such types of information are also of interest in some robotic navigation missions in the application areas of security and surveillance. This paper provides some basic algorithms for determining which spaces the prey can and can not escape from. These algorithms are based on Distance Transform methodology [Jarvis, 1984a]. Simplicity and generality are emphasised and a number of illustrative examples provided. Extensions of the approach are also suggested.

1 Introduction

Given an obstacle strewn field through which both a predator and a prey must navigate without collision, it is of interest to determine which parts of free space afford an opportunity for the prey to escape to a safe haven with the predator in pursuit and how best to achieve this escape. When the obstacle field is initially unknown this information can not be determined 'ab-initio' but is subject to exploration acquired information. This adds a certain complexity to the issue. Whether the position of the predator is known (to the prey) or not or known only in a probabilistic way, also complicates the analysis. It is also fairly obvious that the relative speeds of predator and prey play a critical role and this too must be accommodated. The paper presents the material from the prey's viewpoint, but clearly there is some complementarity in the reverse aspect.

The type of analysis presented in the paper is of relevance to pursuit games whether for entertainment emulation or real time robotic navigation for security/surveillance missions. This work is also part of our research on developing strategies for robots involved in covert navigation missions [Marzouqi and Jarvis, 2003; 2004; 2005] where the ability to escape to a safe area in environments subject to enemy surveillance is an important tool for mission success.

The following sections deal with the underlying methodology used (Distance Transform) and progresses with an outline and example of solutions for simple but general cases. Then follows discussion, including suggestions for further work and finally, the conclusions.

2 The Basic Distance Transform

Distance transforms (DT) were presented in the early image processing literature [Rosenfeld and Pfaltz, 1966] as a means to analyse the shape of binary blobs against a background. Jarvis [1984a] turned the problem inside out by developing an extension of the original DT algorithm to propagate distance through tessellated free spaces around obstacles (like image blobs) for optimal robot path planning purposes. A number of related papers are also of relevance [Jarvis, 1989; 1994]. The algorithm is given in Appendix I which also contains the algorithm for transversal of the optimal path to the goal (steepest descent).

There are a number of useful properties of the approach which can be exploited for various purposes in the path planning domain. Some of these are outlined below.

Firstly, the DT is a space filling distance propagation process which marks all of free space with integers indicating the minimal number of steps to the goal. In this sense, it provides a multi-query solution to optimal path planning from any place in free space to the goal. Secondly, the existence of multiple goals requires no modification of the algorithm, the distance field in such cases indicating the shortest path length to the nearest goal and how to get there. Thirdly, the access cost to any cell from any of its immediate neighbours can be specified non-uniformly as will. Various costs related to ruggedness of the terrain, one way restrictions and other considerations can be easily accommodated [Jarvis, 1999]. Fourthly, the methodology extends to any spatial dimensions without special considerations [Jarvis, 1984b]. Fifthly, time/space extensions are easily accommodated [Jarvis, 1989]. Sixthly, safe paths which maximise the distance to the nearest obstacle can be calculated using the same methodology [Zelinsky, 1994]. With regard to the cost field notions of the fourth property, listed above, fields related to visibility have also been used in 'covert robotics' [Marzouqi and Jarvis, 2003] and in the calculation of visibility itself [Jarvis, 2004].

Here, yet another property is exploited, that of distance relativities amongst entities sharing the same obstacle strewn field. In this way predator/prey situations can be analysed for the purpose of anticipation and strategy (from the viewpoint of the prey in this paper but can be 'reversed' to the predator's viewpoint which is complimentary).

3 Predator/Prey Capture Potential Analysis

Consider, for simplicity, the case of a single predator, a single prey and a single safe haven goal in a completely known obstacle field environment of uniformly tractable free space when the position of the predator is known to the prey and vice versa and both know their own locations. Let both predator and prey be capable of moving at the speed of one cell per time unit.

A DT propagated out from the predator indicates how many steps (and thus time units) are required for it to reach any part of free space. Let A(i,j), i=1,N , j=1,N represent this DT in a rectangularly tessellated space of N x N cells. Obstacle cells are represented by a very large number. Similarly, a DT propagated out from the prey indicates how many steps (and thus time units) are required for it to reach any part of free space. Let B(i,j), i=1,N , j=1,N represent this DT. Once again, obstacle cells are represented by large numbers.

Skipping over obstacle cells, all cells for which A(i,j)< B(i,j),i=1,N,j=1,N are cells the prey can reach before the predator. This set of cells can be regarded as potentially safe to reach but not necessarily to stay at. The complement set can be regarded as unsafe or forbidden and can be treated in the same way as obstacle cells. If the single haven cell exists within the safe set, the prey has an opportunity to escape (but, of course, could fail to take this opportunity). Figure 1 shows an example. The initial locations of both the prey and predator, and two havens' locations are indicated by their names. Obstacles are in black. The calculated safe field is the shaded area. In this case, 'Heaven-1' is considered the only safe haven as it is within the safe zone.

Propagating a third DT out from the haven cell inside the safe set, treating the unsafe set as obstacles, the prey can make an optimal path length (minimal time) retreat to the single safe haven cell simply by following the steepest decent DT trajectory in this third DT field. All cells on the way to the haven can be reached first by the prey. A pursuit game experiment is presented later in Section 4.

If the possible speeds of either the predator or prey are other than one cell per time unit, appropriate scaling of the DT through simple division accounts for this. For example, if the predator can move at 2 steps per time unit, but the prey at only one step per time unit, all the DT values in the propagation out from the predator are simply divided by two. Figure 2 shows a similar case as in Figure 1 but with the predator having twice the speed than the prey. In this case, no havens are guaranteed to be reached by the prey before being captured. Variations in tractorabilty over the field can be also accommodated even if such fields differ between predator and prey. For example, it may be that predator can move more slowly than the prey on smooth terrain but make better progress in rough.

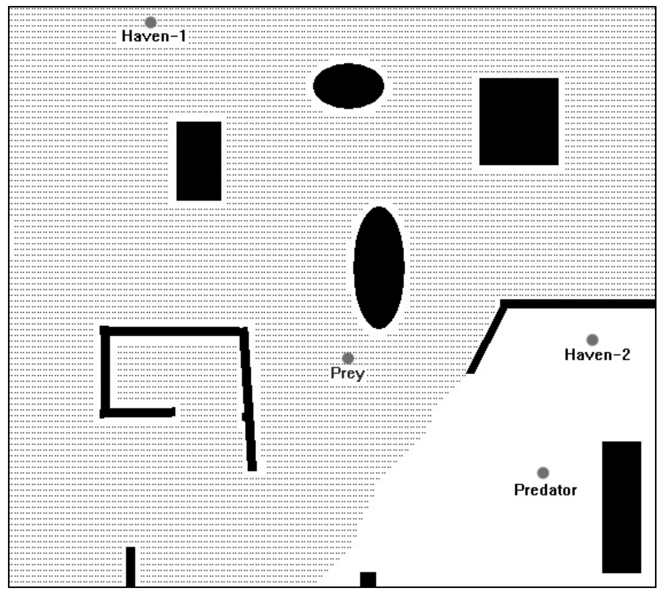


Figure 1. Haven-1 is considered a safe haven as it is within the safe field where the prey can reach anywhere within it before the predator.

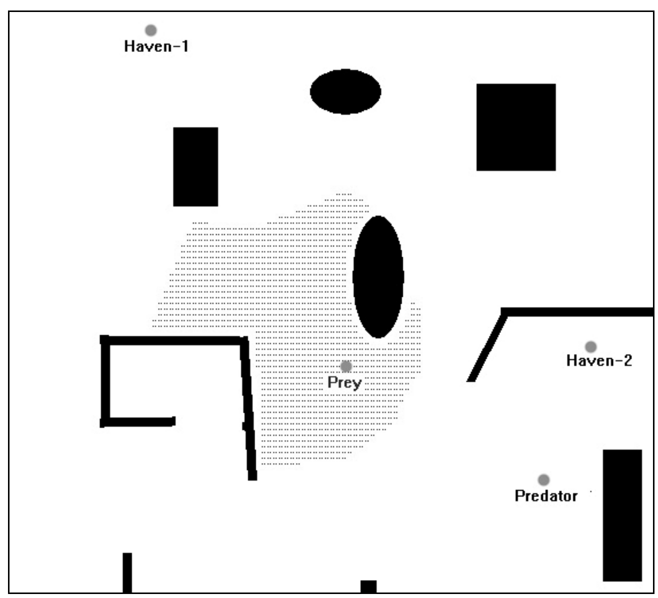


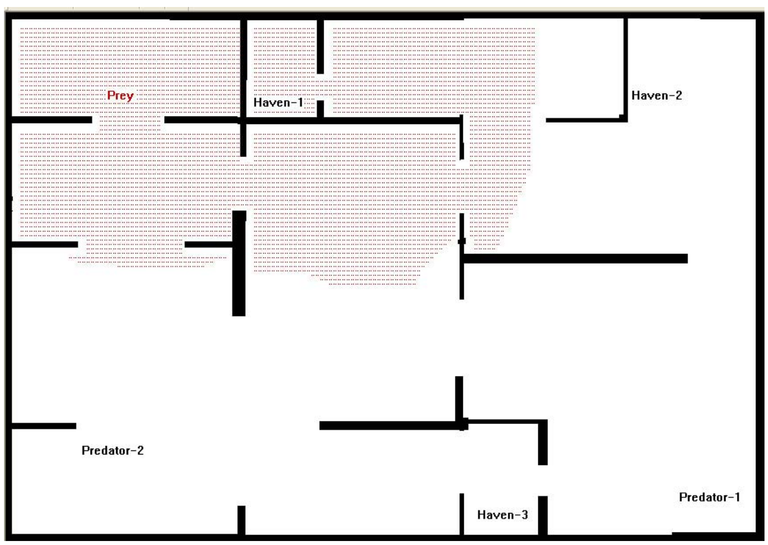
Figure 2. The predator has twice the speed than the prey. The prey's safe field does not include any haven. Hence, reaching any haven before the predator is not guaranteed.

Now suppose there are several safe haven cells. The DT is performed out from those cells in the normal process without any change. The field now indicates the number of steps to the nearest haven cell. Likewise, if there is more than one predator, the relevant DT field indicates where one of them can reach in the specified number of steps. The scaling for speed still holds. If the predators can move at carrying speeds amongst themselves, this will not work, however.

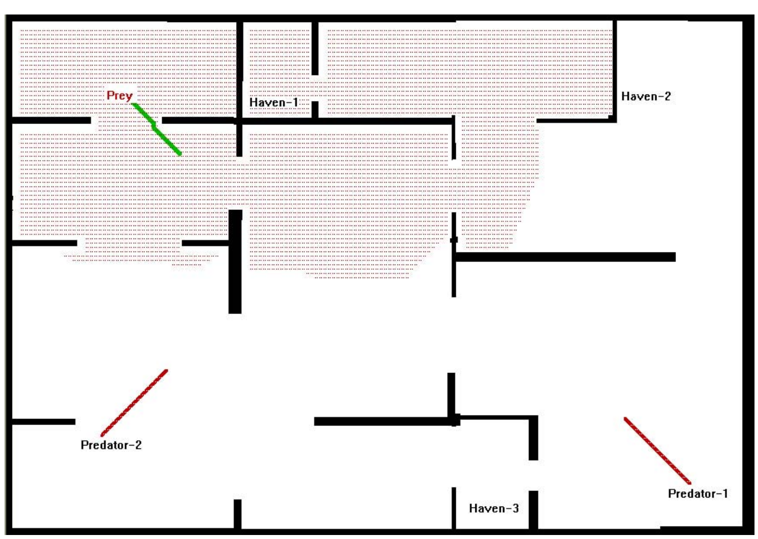
4 Simulation Experiment

A simulated pursuit experiment is carried out to illustrate the above mechanism. When the prey moves towards the nearest haven cell within the 'safe' zone, the predator move towards the prey along the shortest path, but since the predator does not know the location of havens, it has no way of 'cutting off' the retreat in some strategic way. The predator speed relative to the preys is adjustable. When movement occurs the whole cycle of computation is repeated with updated locations. Video sequences showing the dynamics of such experiments will be shown at presentation.

An experiment is presented here in Figure 3. Four 'frozen time' sequences for various time counts are shown. There are two predators and three havens. The two predators' speeds are set to be the same as the prey speed. Figure 3a shows the initial stage of the pursuit where only one haven (Haven-1) is within the prey's safe zone. Figure 3b is a snapshot after 20 cell steps where each entity's location is at the end of its path line. The prey is navigating towards Haven-1. Figure 3c shows clearly how the safe zone shape has changed giving the new entities' locations. The safe zone has included another haven (Haven-2) that was not safe before. At this stage, the prey is navigating towards Haven-2 as it is the closest safe haven. Figure 3d shows the prey has successfully reached Haven-2 before being caught.



(a)



(b)

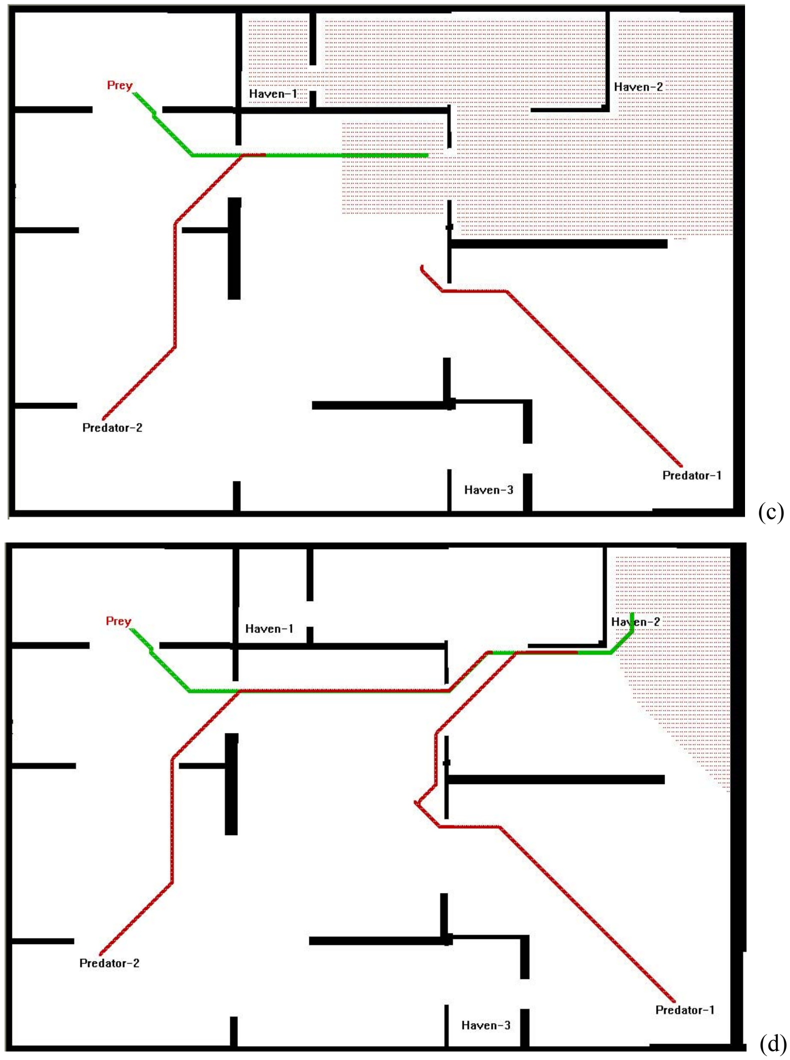


Figure 3. A pursuit experiment of two predators and three havens. a) Only Haven-1 is within the prey's safe zone initially. The prey will escape towards it. b) A snapshot after 20 cell steps where each entity's location is at the end of its path line. c) Haven-2 has become a safe haven. The prey is escaping towards it now as it is the nearest safe haven. d) The prey has escaped successfully.

5 Initially Unknown Obstacle Strewn Environments

Since only the obstacles are responsible for restricting the paths of both predator and prey, their deployment in the environment is critical to this analysis. Since, if both predator and prey can build obstacle maps incrementally, say in a line-of-sight mode, they would do so from different view points so that any instantaneous model of the world held by each would differ until both have explored the whole environment. The more each knows the more effective at their tasks they will be but freedom to explore is predicted by survival strategies on the part of the prey and interception strategies on the part of predator. If both adopt the so called 'optimistic' assumption that all unknown space is regarded as free until shown otherwise, the evaluations considered above can be applied unaltered except for the need to update the analysis as map details develop.

It seems reasonable for the prey to move towards its nearest haven point at each cycle of re-computation of the DT as map changes take place and the known location of the predator changes. The best strategy for a predator is not so simple since so far we have assumed it does not know where the haven cells are. In the experiments of section 4, above, we have simply forced the predators to head towards the prey positions rather than try to intercept the prey in a more effective way. This question will not be covered in this paper and only the prey's situation considered even though the computation is incomplete. Thus the building of maps by the predator will not be considered since the prey does not have access to these maps but only knows the predator's location (which obviously can change). The multiple predator/haven cell case will still be automatically accommodated by the methodology.

6 Discussion and Future Possibilities

What has been covered here is just the beginnings of an analysis tool for pursuit games or robotic escape strategies from the viewpoint of the prey. Developing the appropriate counter strategy needs careful consideration despite complimentarity. Note also that whilst speed can be taken into account the simple DT procedure does not accommodate a variation of speed amongst multiple predators. Since the DT is a multi-query path planning strategy, the solutions provided in the previous two sections apply to any free cell point which may be the position of the prey. The path analysed in regard to incremental discovery of the map should of course relate to a particular choice of initial prey position. Untouched in this paper, yet worthy of future consideration is the possibility of multi robot cooperation on both sides of the competition and probabilistic representation of uncertainty of location of predator(s), prey and/or obstacles. The computational complexity would necessarily explore to include these considerations but this should not preclude such developments.

7 Conclusion

This paper has presented some simple algorithms based on the Distance Transform which shed some light on pursuit game strategies which have application in robotics for surveillance and security. More complex and realistic considerations are worthy of future work and only some simple beginnings of a possible fertile approach are represented here.

References

[Rosenfeld and Pfaltz, 1966] A. Rosenfeld and J.L. Pfaltz, Sequential Operations in Digital Image Processing, *J.A.C.M.*, Vol.13. No.4, Oct, 1966, pp.471-494.

[Jarvis, 1984a] R. A. Jarvis, Collision-free path trajectory using distance transforms, *Proc. National Conference and Exhibition on Robotics*, Melbourne, August 1984.

[Jarvis, 1984b] Jarvis, R.A. Projection Derived Space Cube Scene Models for Robot Vision and Collision Free Trajectory Planning, invited paper *2nd International Symposium of Robotics Research*, Kyoto, Japan, August 1984. M.I.T. Press, 1985 pp.403-410.

[Jarvis, 1989] Jarvis, R.A. Collision-Free Path Planning in Time Varying Environments, *Proc. I.E.E.E./R.S.J. International Workshop on Intelligent Robots and Systems, '89*, Tsukuba, Japan, Sept. 4-6 '89, pp 99-106.

[Jarvis, 1994] Jarvis, R.A. On Distance Transform Based Collision-Free Path Planning for Robot Navigation in Known, Unknown and Time-Varying Environments, invited chapter for a book entitled '*Advanced Mobile Robots*' edited by Professor Yuan F. Zang World Scientific Publishing Co. Pty. Ltd. 1994, pp. 3-31. 8.

[Jarvis, 1999] Jarvis, R.A. Autonomous Navigation of a Martian Rover in Very Rough Terrain, *Proc. International Symposium on Experimental Robotics,* March 26-28, 1999, Sydney University, pp.225-234.

[Jarvis, 2004] R. A. Jarvis, “Distance transform based visibility measures for covert path planning in known but dynamic environments,” *Proc. 2nd International Conference on Autonomous Robots and Agents* ICARA, New Zealand, December, 2004.

[Marzouqi and Jarvis, 2003] Mohamed Marzouqi and Ray A. Jarvis, Covert path planning for autonomous robot navigation in known environments, *Proc. Australasian Conference on Robotics and Automation*, Brisbane 2003.

[Marzouqi and Jarvis, 2004] Mohamed Marzouqi and Ray A. Jarvis, Covert Robotics: Hiding in Known Environments, *Proc. 2004 IEEE Conference on Robotics, Automation and Mechatronics* (RAM), Singapore, December 2004.

[Marzouqi and Jarvis, 2005] Mohamed Marzouqi and Ray A. Jarvis, “Covert path planning in unknown environment with known or suspected sentry location,” *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems* (IROS), Canada, August 2005.

[Zelinsky, 1994] A. Zelinsky. “Using path transform to guide the search for findpath in 2d,” *International Journal of Robotics Research*, vol. 13(4), pp. 315-325, August 1994.

Appendix I

(a) Distance Transform Algorithm:

```
(*Initialize cell & blocked border around cell at
 x=0, xmax+1 & y=0, ymax+1*)
for y:=0 to yMax+1 do
for x:=0 to xMax+1 do
    if goal [x,y] then cell [x,y]:=0;
    else cell; [x,y]:=xMax*y Max; (*A large number *)
(*Calculate distance transform in cell [ ]*)
repeat
for y:=2 to yMax do
for x:=2 to xMax do
    if not blocked [x,y] then
        cell [x,y]:= min (cell[x-1,y]+1, cell[x-1,y-1]+1, cell [x,y-
            1]+1, cell[x+1,y-1]+1, cell [x,y]);
for y:=yMax-1downto 1 do
for x:=xMax-1 downto do
    if not blocked [x,y] then
    cell[x,y]:=min(cell[x+1,y]+1,cell[x+1,y+1]+1,
        cell[x,y+1]+1,cell[x-1,y+1]+1,cell[x,y]);
until no change;
```

(b) Shortest Path Trajectory Algorithm:

```
(*Trace all paths*)
for ys:= 1 to yMax do
for xs:= 1 to xMax do
    if start (xs,ys) then
        (*Trace path to nearest goal for start point (xs,ys)*)
        begin
        x:=xs; y:=ys;
        while not goal [x,y] do
            begin
            next(x,y,xn,yn);
            x:=xn;y:=yn;
            end;
        end;
```