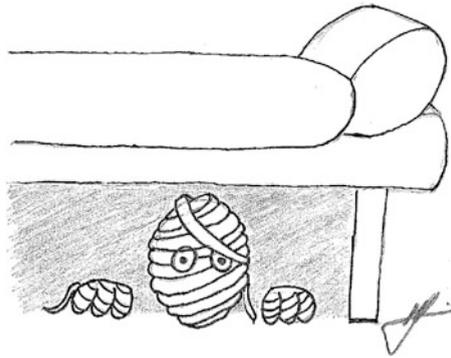


# Chapter 2

## Search Games for an Immobile Hider

Thomas Lidbetter



**Abstract** A search game for an immobile hider is a zero-sum game taking place in some search space. The hider picks a point in the space and a searcher who is unaware of the hider’s location moves around attempting to find him in the least possible time. We give an overview of the theory of search games on a network with an immobile hider, starting with their conception in the Rufus Isaac’s 1965 book on Differential Games, then moving on to some classic results in the field from Shmuel Gal and others. Finally we discuss some recent work on new search game models which consider, for example, what happens when the searcher does not have a fixed starting point or when the speed of the searcher depends on the direction in which he is traveling.

---

T. Lidbetter (✉)  
Department of Mathematics, London School of Economics, London, UK  
e-mail: [t.r.lidbetter@lse.ac.uk](mailto:t.r.lidbetter@lse.ac.uk)

## 2.1 Introduction

Since the conception of *search games* almost 50 years ago, the field has expanded and developed in many different directions, as seen in Chap. 1. In this chapter we focus in on one particular theme: that of search games on a network with a mobile searcher and an immobile hider. Games of this type may be described as ‘hide-and-seek’ games. The classic results in this field can be found in Alpern and Gal’s monograph [4] and Gal’s recent survey [13]. Here we do not aim to give an exhaustive list of all work in the field, but we follow on from Sects. 1.1.1 and 1.1.2 in Chap. 1, taking a more detailed look at some classic results and linking them to new work on search games with an immobile hider.

We begin in Sect. 2.2 by discussing how Isaacs [14] first introduced search games of this type, and how he described strategies for both the hider and the searcher which would continue to be of fundamental importance in later work in the field. In Sect. 2.3 we then turn to the first rigorous definition, given by Gal [10], of a search game with an immobile hider and a mobile searcher who starts from a given point. We indicate how Gal solved his game if the search space is a tree or if it is Eulerian.

We then show in Sect. 2.4 how Reijnierse and Potters [17] extended Gal’s analysis to *weakly cyclic* networks, which have the structure of a tree with some nodes replaced by cycles. We describe the solution of Gal’s game on these networks, and how Gal proved an analagous result for weakly Eulerian networks.

In the final two sections we discuss some more recent work on search games on networks with an immobile hider. Section 2.5 deals with a version of Gal’s original game in which the searcher can start from any point in the network. Section 2.6 describes three new Search Game models [2, 5, 6] which all modify or generalize Gal’s classic model in some way.

## 2.2 The Birth of Search Games

Search games were first introduced by Rufus Isaacs in his 1965 book *Differential Games* [14], as indicated in Chap. 1. The book was originally motivated by combat problems, and indeed, many of the problems discussed in the book have a military focus to them. Earlier chapters in the book are concerned with so called *Pursuit Games*, in which a *Pursuer* (or *Pursuers*) aim to capture an *Evader* whose location is known to him at all times during the game. Search games are introduced later in the book in a chapter called ‘Toward a Theory with Incomplete Information’. The model presented differs from Pursuit Games in that Pursuers now aim to capture an Evader about whose position the Pursuers now do not have complete information. The terminology changes: the Evader becomes the hider and the Pursuers become the searchers. This terminology has stuck and is now widely used in the search games literature.

Isaacs begins by defining what he calls the *simple search game*. This could be regarded as the simplest and most general possible search game, and is described in informal terms. In an arbitrary region  $\mathcal{R}$ , which may be a subset of Euclidean space

of any dimension, a hider picks a hiding point (that is a point in  $\mathcal{R}$ ). The searcher then picks some sort of unit speed trajectory in the region. The payoff (or search time) is the time taken until the searcher's trajectory meets the hider. There is an assumption that the searcher is able to find a tour of the region that is not wasteful, so that it does not 'double back' on itself. The solution of the game Isaacs gives is simple: the searcher picks one such tour  $S$ , then follows it with probability one half and follows the reverse tour with probability one half. Supposing  $\mathcal{R}$  has measure  $\mu$ , if  $S$  finds a point in  $\mathcal{R}$  at time  $t$ , the reverse of  $S$  will find the same point at time  $\mu - t$ . Hence the expected time  $T$  to find any given point is given by

$$T = 1/2t + 1/2(\mu - t) = \mu/2$$

The value of the game is therefore at most  $\mu/2$ . The hider can ensure the payoff is no more than  $\mu/2$  by hiding uniformly in  $\mathcal{R}$ , so that the probability he hides in any subset of  $\mathcal{R}$  is proportional to its measure. By using this strategy, the hider ensures that the probability the searcher finds him before time  $t$  is no more than  $t/\mu$  for  $0 \leq t \leq \mu$ , so the probability the search time is  $t$  or more is at least  $1 - t/\mu$ . Hence the expected time  $T$  satisfies

$$\begin{aligned} T &= \int_0^\infty Pr(\text{search time is } \geq t) dt \\ &\geq \int_0^\mu (1 - t/\mu) dt \\ &= \mu/2. \end{aligned}$$

The value of the game is therefore at least  $\mu/2$ , and combining the bounds we have

**Theorem 1 (Isaacs).** *The value of the simple search game is  $\mu/2$ .*

These strategies given by Isaacs are important and direct a lot of the later research on search games.

## 2.3 Search Games on Networks

A more precise formulation of Isaac's game is given by Gal [10] and [11]. Gal focuses on the game played on a network  $Q$ , which is any connected finite set of arcs of measure  $\mu$  with a distinguished starting point  $O$ , called the root. The hider picks a point  $H$  in  $Q$  and the searcher picks a unit speed path  $S$  starting from  $O$ . The payoff (or search time) is the time taken for the path to reach  $H$ . This game is mentioned in Sect. 1.1.1 of Chap. 1.

In [10], Gal uses Isaacs hider strategy to give a lower bound for the value  $V$  of the game: by hiding uniformly in the network the hider can ensure that the search time always at least  $\mu/2$ . We call this strategy  $u$ . However, the assumption made

by Isaacs that the searcher can find a non-wasteful trajectory is not made, so the searcher strategy given in [14] is not always available and the value of the game may be greater than  $\mu/2$ . The searcher is also restricted to picking a path which starts from  $O$ , so it may not be possible for him to implement the ‘reverse’ of a path. For instance, if  $Q$  is a single unit length arc with the root  $O$  at one end and a point  $A$  at the other, the value of this game is clearly  $1 > \mu/2$ . The hider simply uses the pure strategy of hiding at  $A$  and the searcher picks the path from  $O$  to  $A$ .

However, adapting the searcher strategy given in [14], Gal gives an upper bound for the value. The searcher may not be able to find a non-wasteful, reversible path in  $Q$ , but he will always have some minimal time tour  $S$  of  $Q$  starting and ending at  $O$  of length  $\bar{\mu} \geq \mu$ . He can then use the mixed strategy where he picks  $S$  with probability  $1/2$  and the reverse of  $S$  with probability  $1/2$ , ensuring that he finds every point in  $Q$  in expected time no more than  $\bar{\mu}/2$ . The searcher’s minimal tour  $S$  is later called a Chinese Postman Tour (CPT) in [12], and the randomized strategy given here is called the Random Chinese Postman Tour (RCPT). The RCPT gives an upper bound for the value  $V$ , and combining this with the lower bound we have

$$\mu/2 \leq V \leq \bar{\mu}/2 \tag{2.1}$$

Gal examines when these two bounds are tight. Suppose  $Q$  is Eulerian, so that it has a continuous closed path that visits each point of  $Q$  exactly once. Then the searcher’s CPT is one such Eulerian path starting at  $O$ . Since the length  $\bar{\mu}$  of this tour is  $\mu$ , the bounds in (2.1) are tight and we have  $V = \mu/2 = \bar{\mu}/2$ . The uniform strategy  $u$  is optimal for the hider. It is easy to see that Eulerian networks are the only networks for which  $\bar{\mu} = \mu$ .

We can also consider the game played on a tree, that is a network without any cycles. In a sense, a tree is the opposite of an Eulerian network since the CPT of a tree has the maximum possible length,  $\bar{\mu} = 2\mu$ , as all arcs must be traversed in both directions. The inequalities (2.1) therefore become  $\mu/2 \leq V \leq \mu$ . Clearly the uniform hider strategy  $u$  is not optimal for the hider, since every point  $H$  of  $Q$  is dominated in strategies by a leaf node (a node of degree 1). Hence an optimal hider strategy must be some distribution on the leaf nodes. In [10] Gal defines a hider distribution later called the Equal Branch Density (EBD) distribution in [12], and shows that it is optimal for the hider, guaranteeing him an expected search time of no less than  $\mu = \bar{\mu}/2$ , which is the value of the game. The RCPT is optimal for the searcher.

The EBD distribution can be defined in terms of a concept called *search density*, which extends to general search spaces  $Q$  that may not be networks. For a connected subset  $A$  of  $Q$  and a hider hidden on  $Q$  according to a fixed distribution, the search density  $\rho(A)$  is defined as the time taken for the searcher to tour  $A$  divided by the probability the hider is in  $A$ . Consider a tree  $Q$  and a node  $x$  of  $Q$  that has degree at least 3. We call  $x$  a branch node. The arcs touching  $x$  consist of one arc on the path from  $x$  to  $O$  and some other arcs. For each of these other arcs  $a$ , we define a branch at  $x$  which consists of  $a$  together with all arcs whose unique path to  $O$  intersects with  $a$ . The EBD distribution is the unique hider distribution on the leaf nodes of  $Q$  that ensures that at every branch node of  $Q$ , all branches have equal search density.

We illustrate the EBD distribution with an example. In Fig. 2.1 nodes are labelled by letters and arc lengths indicated by numbers. To calculate the EBD distribution on this network, first note that there are two branches at  $O$ , which must have equal search density. This can be achieved by assigning hider probability  $3/9 = 1/3$  to the branch consisting of the arc  $OC$ , and probability  $2/3$  to the other branch. The branch node  $D$  has two branches, and to ensure these have equal search density, the hider probability assigned to the arcs  $AD$  and  $BD$  must be proportional to 2 and 3, respectively. Hence the probabilities the hider is at nodes  $A$  and  $B$  are  $2/5 \cdot 1/3 = 2/15$  and  $3/5 \cdot 1/3 = 3/15$  respectively. The probability the hider is at  $C$  is  $1/3$ .

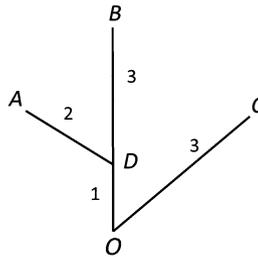


Fig. 2.1 A tree network

In [10], Gal shows that if the hider uses the EBD distribution, this ensures that any depth-first search of  $Q$ , and in particular any CPT finds the hider in expected time exactly  $\mu = \bar{\mu}/2$ , which must therefore be the value of the game. In the case of the network in Fig. 2.1, the value of the game is  $\mu = 9$ .

Hence we have

**Theorem 2 (Gal).** *If  $Q$  is an Eulerian network or a tree then the value of the search game with an immobile hider played on  $Q$  is  $\bar{\mu}/2$ .*

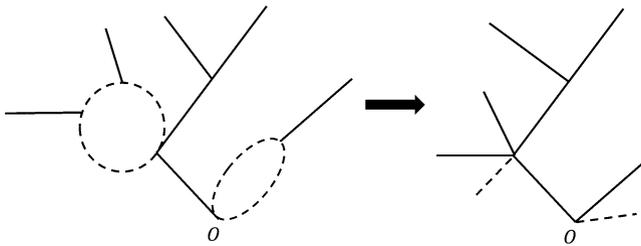
As discussed in Sect. 1.1.1 of Chap. 1, the RCPT is not optimal for all networks, in particular the 3-arc network depicted in Fig. 1.1, though this was not shown for another 15 years [15].

## 2.4 Weakly Cyclic and Weakly Eulerian Networks

Solutions of the game described in the previous section are not limited to trees and Eulerian networks. In [17] Reijniere and Potters solve the game for *weakly cyclic* networks, showing that the RCPT is optimal for the searcher, so that the value is  $\bar{\mu}/2$ . A weakly cyclic network can be thought of as a tree network for which some of the nodes have been replaced with cycles. Alternatively, a weakly cyclic network can be defined more precisely as a network for which there are at most two disjoint paths between any two nodes. Weakly cyclic networks cannot contain any subnetwork

that is topologically homeomorphic to the three arc network depicted in Fig. 1.1. A weakly cyclic network is depicted on the left hand side of Fig. 1.2; the cycles are indicated by the dotted lines.

Reijnierse and Potters give an algorithm to calculate the optimal hider distribution, in which the hider hides with some probability on leaf nodes and with some probability hides uniformly on the cycles. Alpern and Gal [4] later give an alternative version of the algorithm, in which every cycle in the network is replaced with a leaf arc of half the length of the cycle, and the EBD distribution is calculated on the new network. The network depicted on the right hand side of Fig. 1.2 is the modification of the weakly cycle network on the left. The hider probability that should be assigned to a cycle in the original network is then the probability assigned to the end of the associated leaf arc in the new network (Fig. 2.2).



**Fig. 2.2** A weakly cyclic network and its modifications

Reijnierse [16] later showed that the equivalent result holds if we replace ‘weakly cyclic’ with ‘weakly Eulerian’. A network is weakly Eulerian if it can be obtained from a tree by replacing some nodes with Eulerian networks. Gal [12] found a simple proof of this result, showing not only that the value  $V$  of the game is  $\bar{\mu}/2$  for weakly Eulerian networks, but, as mentioned in Chap. 1, these are the only networks for which this is the value, and the RCPT is optimal. In summary,

**Theorem 3 (Gal).** *The value of the search game with an immobile hider played on a network  $Q$  is  $\bar{\mu}/2$  if and only if  $Q$  is weakly Eulerian.*

Notice that the class of weakly Eulerian networks includes both trees and Eulerian networks, so Theorem 3 generalizes Theorem 2.

## 2.5 Search Games Without a Fixed Searcher Starting Point

In a recent paper [9] Dagan and Gal define a new Search Game model on a network  $Q$  in which the assumption that the searcher has a fixed starting point  $O$  is dropped, and the searcher can begin his search from any point on  $Q$ . This model has already been discussed in Sect. 1.1.2 in Chap. 1, where it was noted that provided the searcher has some Eulerian path (one which visits every point of the network exactly

once), Isaac's result holds and the value of the game is  $\mu/2$ . The searcher can simply choose the Eulerian path with probability  $1/2$  and its reverse with probability  $1/2$ ; the hider can hide uniformly on the network. The networks that have an Eulerian path include the 3 arc network in Fig. 1.1, whose solution in Gal's classic model was so elusive. For the arbitrary start model, the value of this game is  $\mu/2 = 3/2$ . Just as we define Chinese Postman Tours, we can define a *Chinese Postman Path* of a network  $Q$  as a minimal time path that visits all the points of  $Q$ . We can then define  $\tilde{\mu}$  as the length of a Chinese Postman Path, and we obtain a result analogous to (2.1) for the value  $V$  of the Search Game played on networks with an arbitrary starting point:

$$\mu/2 \leq V \leq \tilde{\mu}/2. \quad (2.2)$$

The arbitrary start model was further studied in [3] in which the authors call a network *simply searchable* if the upper bound on  $V$  in (2.2) is tight. They give sufficient conditions for a network to be simply searchable, and in particular they show that trees are simply searchable and that the hider should use the EBD distribution, with respect to a root located at the *center* of the tree: that is the point  $c$  whose greatest distance from any other point in the tree is minimal. For example, in Fig. 2.1 the center  $c$  is located halfway between nodes  $O$  and  $D$ . If we add a node at  $c$ , then at this point there are two branches of lengths  $7/2$  and  $11/2$ , which the hider chooses with probabilities proportional to 7 and 11, respectively. Hence the hider chooses the node  $C$  with probability  $7/18$  and nodes  $A$  and  $B$  with total probability  $11/18$ .

## 2.6 Other Search Game Models

Recently some alternative models of search games on networks have been proposed. In the models we have discussed so far the searcher's strategy space is a set of unit speed paths. However we might consider associated games in which the searcher has a different strategy set.

In [2] Alpern defines a new model called *find-and-fetch* in which he considers a searcher who not only wishes to find a hider but also wishes to return to the root  $O$ . This models common problems such as search-and-rescue and foraging problems in which an animal must find food and then return to its lair. As in Gal's model, the searcher follows a unit speed path from  $O$ , but then upon reaching the hider takes the shortest path back to  $O$  at speed  $\rho$ . The payoff is the total time to find the hider and return to  $O$ . In the case of a bird being weighed down by food he is taking back to his nest we might have  $\rho < 1$ , whilst  $\rho > 1$  might be more appropriate for the case of someone searching for a contact lens, in which the return speed would be quicker.

Alpern finds that if  $Q$  is a tree, the optimal strategy for the hider is still the EBD distribution in this game. However, the RCPT is no longer optimal for the searcher. Instead, he randomizes between all possible depth-first searches using a type of strategy called a branching strategy. Upon reaching a node for the first time the searcher chooses which outward branch to take according to a certain probability. Alpern proves the following.

**Theorem 4 (Alpern).** *The value  $V$  of the find-and-fetch game on a tree is*

$$V = \mu + D/\rho, \quad (2.3)$$

where  $D = D(Q)$  is the mean distance from  $O$  to the leaf nodes of  $Q$ , weighted according to the EBD distribution.

To illustrate how  $D$  is calculated, consider the network in Fig. 2.1. The probabilities that the hider is at nodes  $A$ ,  $B$  and  $B$  are  $2/15$ ,  $3/15$  and  $10/15$ , respectively, and the distances of these nodes from  $O$  are 3, 4 and 3. Hence  $D = 2/15(3) + 3/15(4) + 5/15(3) = 11/5$ . For  $\rho = 1/2$ , the value  $V$  of the find and fetch game played on the tree in Fig. 2.1 is  $V = 9 + (11/5)/(1/2) = 10.1$ . Note that as  $\rho$  tends to infinity so that the searcher can return instantaneously to the root after finding the hider, the value  $V$  in (2.3) tends to  $\mu$ , Gal's classic result (Theorem 2).

A different model of search is given in [6], in which the authors suppose that the searcher can use an *expanding search*. This is defined as a sequence of unit speed paths on a network  $Q$ , starting at  $O$ , each of which starts from a point already reached by the searcher. Another way to think of expanding search is as a family of connected subsets of  $Q$  starting with  $O$  and expanding at unit speed. To differentiate expanding search from the type of search used in Gal's model, we call the latter pathwise search. Expanding search provides a model of mining, in which the time taken to recommence mining from a location already reached is small compared to the time taken up by the mining itself. As before, the hider simply picks a point on  $Q$  and the searcher picks an expanding search. The search time is the time taken for the searcher to reach the hider.

Again, if  $Q$  is a tree it turns out that the EBD distribution is optimal for the hider, and the searcher's optimal strategy is a branching strategy. The authors show that

**Theorem 5 (Alpern and Lidbetter).** *The value  $V$  of the expanding search game on a tree is*

$$V = 1/2(\mu + D), \quad (2.4)$$

The variable  $D$  is defined as above. In the case of the network in Fig. 2.1 where  $D = 11/5$  and  $\mu = 9$ , the value is  $V = 1/2(9 + 11/5) = 5.6$ .

In [6] the expanding search game is also solved for 2-arc-connected networks. These are networks that cannot be disconnected by the removal of fewer than two arcs. The authors show that on these networks it is optimal for the hider to hide uniformly, and for the searcher to make an equiprobable choice of a reversible expanding search and its reverse. A reversible expanding search is simply one whose reverse is also an expanding search, analogous to an Eulerian circuit in Gal's model. The authors show that such a search always exists on a 2-arc-connected network, and the randomized choice of this search and its reverse ensures that the searcher finds the hider in expected time no greater than  $\mu/2$ , which is the value of the game. For example, the 3-arc network depicted in Fig. 1.1 in Chap. 1 is 2-arc-connected, and hence has value  $\mu/2 = 3/2$ .

For trees, the find and fetch model and the expanding search model can both be encapsulated in a single overarching model. In [1] Alpern examines the Search Game on a network with asymmetric travel times, meaning that the speed it takes for the searcher to traverse an arc depends on the direction in which he travels. An equivalent formulation is given in [5] in which the searcher moves with a speed that depends on his direction of travel. We therefore call this the *variable speed* model. The game is then defined as usual: the searcher picks a path in the network starting at  $O$ , the hider picks a point on the network and the payoff is the time taken for the searcher to reach the hider. The model clearly encompasses Gal's model of networks with symmetric travel times if the travel times of each arc are set to be the same in either direction.

For a tree  $Q$  we can give every point  $x$  on  $Q$  a height  $\delta(x)$ , equal to the time taken to travel from  $O$  to  $x$  (along the shortest path) minus the time taken to travel from  $x$  to  $O$ . This definition is motivated by the assumption that it is quicker to travel uphill than downhill. In [1] Alpern shows that the EBD is once again optimal for this game played on a tree, and he gives recursive formulae for the optimal branching strategy for the searcher. In [5], the authors derive a closed form expression for the optimal searcher strategy as well as a formula for the value  $V$  of the game:

**Theorem 6 (Alpern and Lidbeter).** *The value  $V$  of the variable speed search game is*

$$V = 1/2(\tau + \Delta), \quad (2.5)$$

where  $\tau$  is defined as the time taken for the searcher to tour the network, and  $\Delta = \Delta(Q)$  is defined as the mean height of the leaves, weighted with respect to the EBD distribution.

If the network is symmetric, then all leaf nodes have height 0, and  $\tau = 2\mu$ , so (2.5) reduces to Gal's classic result,  $V = \mu = \bar{\mu}/2$  given in Theorem 2. In fact, in the case that the network  $Q$  is a tree, the variable speed network model also encompasses both the find and fetch model and the expanding search model, as we now explain.

We first consider the find and fetch game, in which the searcher must return to  $O$  along the shortest path at speed  $\rho$  after finding the hider. It is optimal for the hider to choose a leaf node  $x$ , and for any such choice of  $x$  at shortest distance  $d(x, O)$  from  $O$ , the searcher must travel for additional time  $d(x, O)/\rho$  after finding the hider. We therefore form a new network  $Q'$  from  $Q$  by adding an asymmetric arc from  $x$  to a new leaf node  $x^+$  with forward travel time (from  $x$  to  $x^+$ ) of  $d(x, O)/\rho$  and backward travel time  $-d(x, O)/\rho$ . The variable speed game played on  $Q'$  is then equivalent to the find and fetch game played on  $Q$ : traveling to  $x^+$  in  $Q'$  is equivalent to traveling to  $x$  in the original network and then back to  $O$  at speed  $\rho$ , and if the hider is not at  $x$  the extra arc from  $x$  to  $x^+$  makes no contribution to the search time. Hence the two models are equivalent.

The total tour time  $\tau$  of  $Q'$  is equal to twice the length  $2\mu$  of  $Q$ , and in the  $Q'$  the leaf node  $x^+$  has height  $2d(x, O)/\rho$ , so  $\Delta = \Delta(Q')$  is the mean value of  $2d(x, O)/\rho$ ,

weighted with respect to the EBD distribution, which is equal to  $2D(Q)/\rho$ . Hence by (2.5), the value is

$$\begin{aligned} V &= 1/2(2\mu + 2D/\rho) \\ &= \mu + D/\rho, \end{aligned}$$

as given in (2.3).

We now return to the expanding search model played on a tree  $Q$ . Suppose we form a new network  $Q''$  by replacing each arc of  $Q$  of length  $\lambda$  with an asymmetric arc with forward travel time (away from  $O$ ) of  $\lambda$  and backward travel time 0. Then a depth-first pathwise search on  $Q''$  is equivalent to an expanding search on  $Q$ . It can be shown that it is optimal to use a depth-first search in the expanding search game, so that the two models are equivalent. The total tour time of the new network is the length  $\mu$  of the original network, and the height of a leaf node in the new network is the distance from that node to  $O$  in the old network, so  $\Delta(Q'') = D(Q)$ . Hence, by (2), the value is

$$V = 1/2(\mu + D),$$

as given in (2.4).

## 2.7 Conclusion

We have seen how an idea in [14] sparked a field of research which has produced many elegant results, and continues to develop and expand. We have focused here on search games on a network with an immobile hider, but search games are not limited to this paradigm. Much has been achieved in the field of search games with a mobile hider (also originally motivated by Isaacs [14]), as well as many other variations on the classic models. The connected field of search games in unbounded domains, initiated independently by Bellman [8] and Beck [7], has also been extensively studied. Many unanswered questions in search games remain and new problems arise, capturing the imaginations of those who have taken the childhood game of hide-and-seek to its mathematical extreme.

## References

1. Alpern, S. (2010). Search games on trees with asymmetric travel times. *SIAM J. Control Optim.* 48, no. 8, 5547–5563.
2. Alpern, S. (2011). Find-and-fetch search on a tree. *Operations Research* 59 (2011): 1258–1268.
3. Alpern, A., Baston, V. and Gal, S. (2008). Network Search Games with an immobile hider without a designated searcher starting point, *Int. J. Game Theory* 37, 281–302.
4. Alpern S. and Gal S. (2003). *The Theory of Search Games and Rendezvous* (Kluwer International Series in Operation Research and Management Sciences, Kluwer, Boston).

5. Alpern, S. and Lidbetter, T. (2013) Searching a Variable Speed Network. *Mathematics of Operations research* (in press).
6. Alpern, S. and Lidbetter, T. (2013). Mining Coal or Finding Terrorists: the Expanding Search Paradigm. *Operations Research* (in press).
7. Beck, A. (1964). On the linear search Problem. *Naval Res. Logist.* 2, 221–228
8. Bellman, R. (1963). An optimal search problem. *SIAM Rev.* 5, 274.
9. Dagan, A. and Gal, S. (2008). Networks Search Games with Arbitrary searcher starting point, *Networks* 52, 156–161.
10. Gal, S. (1979). Search games with mobile and immobile hider. *SIAM J. Control Optim.* 17, 99–122.
11. Gal, S. (1980). *Search Games*. Academic Press, New York.
12. Gal, S. (2000). On the optimality of a simple strategy for searching graphs. *Int. J. Game Theory* 29, 533–542.
13. Gal, S. (2011). Search games. *Wiley Encyclopedia of Operations Research and Management Sci.* (James J. Cochran, ed.), Wiley.
14. Isaacs, R. (1965). *Differential Games*. Wiley, New York.
15. Pavlovic, L. (1993). Search game on an odd number of arcs with immobile hider. *Yugosl. J. Oper. Res.* 3, no. 1, 11–19.
16. Reijnierse, J. H. (1995). *Games, graphs and algorithms*. Ph.D Thesis, University of Nijmegen, The Netherlands.
17. Reijnierse, J. H. and Potters, J. A. M. (1993). Search games with immobile hider. *Int. J. Game Theory* 21, 385–394.



<http://www.springer.com/978-1-4614-6824-0>

Search Theory

A Game Theoretic Perspective

Alpern, S.; Fokkink, R.; Gąsieniec, L.A.; Lindelauf, R.;

Subrahmanian, V.S. (Eds.)

2013, VIII, 303 p., Hardcover

ISBN: 978-1-4614-6824-0