A geometric analysis of heuristic search

by GORDON J. VANDERBRUG
University of Maryland
College Park, Maryland

ABSTRACT

Search spaces for various types of problem representations can be represented in one quadrant of the coordinate planes. This geometric representation is used to prove some formal properties of heuristic search strategies involving completeness, admissibility, optimality, consistency, and the use of the perfect heuristic. The geometric analysis provides an intuitive alternative to the algebraic analysis which appears in the literature.

INTRODUCTION

To specify a state-space representation for a problem, one must specify a start state, the general structure of a state, a characterization of a goal state, and a set of operators which map states into states. To solve a problem using such a representation one successively applies operators to currently generated states to obtain new states until a goal state is generated. This process is called searching the search space (or state-space), and can be done in many different ways. The manner in which it is done is called a search strategy. The solution to the problem is the sequence of operators which transforms the start state into the generated goal state.

The graph model for searching a state-space is based on associating states in the representation with nodes of the graph, and operators of the representation with arcs of the graph. With this model we can view a state-space representation as an implicit definition of a graph. It defines a start node and procedures for generating other nodes of the graph.

A search strategy can be thought of as a process of making explicit part of this implicitly defined graph. In this paper we will be dealing with a frequently occurring class of search strategies called ordered search strategies. An ordered search strategy determines the manner in which the state-space is generated by assigning a merit ordering to the nodes. A merit ordering is a procedure for ranking the nodes. The search process occurs in stages, and at each state the merit ordering specifies which node is to be expanded (i.e., which node is to have its successor nodes generated by applying all applicable operators). An ordered search algorithm is presented in Figure 1. It is equivalent to the algorithms presented in Hart, Nilsson, and Raphael; Pohl; and Kowalski.

Usually merit orderings are defined by evaluation functions. Evaluation functions use selected features of a state to assign it a number, and thus rank it relative to all the other states. One possible evaluation function assumes that the operators have associated costs, and assigns to each node the sum of the costs of the operators used to generate it. This evaluation function is usually denoted by g, and the search resulting from its use is called the uniform cost search strategy. In terms of the graph which represents the
state-space, the uniform cost strategy uses the cost of the path from the start node to the node \( n \) as the merit of \( n \). Pure heuristic search uses and estimates, \( h(n) \), of the cost of a path from \( n \) to the nearest goal node as the merit of \( n \). Frequently \( h(n) \) is based on the extent to which selected features of the node \( n \) differ from these same features of a goal node. This cost is frequently referred to as the distance from \( n \) to a goal.

We will insist that \( h(n) = 0 \) whenever \( n \) is a goal node. An ordered search strategy which uses an evaluation function \( f(n) = g(n) + h(n) \), with both a cost and a heuristic component, is called diagonal search. Various weights on the cost and heuristic components can be achieved with the function \( f(n) = (1-\omega)g(n) + \omega h(n) \), for \( \omega [0,1] \). \( \omega = 0 \) and \( \omega = 1 \) are uniform cost and pure heuristic search respectively; while \( \omega = \frac{1}{2} \) is diagonal search, since the two evaluation functions, \( \frac{1}{2}g(n) + \frac{1}{2}h(n) \) and \( g(n) + h(n) \), define the same merit ordering.

**DEFINITIONS**

An ordered search strategy is said to be complete if whenever there exists a solution to the problem the strategy will find one. An admissible strategy is one which terminates with a minimum cost solution whenever one exists. The concept of optimality applies to strategies which are admissible and is defined as follows. Let \( h_s \) and \( h_h \) be two heuristic functions such that \( h_s(n) <= h_h(n) \) for all nongoal nodes \( n \), where \( h_p \) is a perfect heuristic function (the heuristic which gives the exact cost to the nearest goal). An admissible strategy is said to be optimal if searching with \( h_s \) expands all of the nodes that searching with \( h_h \) expands. Admissibility can be viewed as the optimality of the solution, whereas optimality is really the optimality of the search process.

In order to prove that ordered search strategies possess the above properties certain assumptions on the heuristic function and the graph of the search space must be made. A heuristic \( h \) satisfies the lower bound condition if \( h(n) <= h_p(n) \) for all \( n \), where \( h_p \) is the perfect heuristic. A heuristic \( h \) is said to be consistent if any \( n \) and \( n' \) such that there is a path from \( n \) to \( n' \), \( h(n) - h(n') <= k(n,n') \), where \( k(n,n') \) is the cost of the path from \( n \) to \( n' \). The concept of a \( \delta \)-graph will also be used in the theorems. We will define a \( \delta \)-graph to be a graph which does not contain a path with an unbounded sequence of partial sums of arc costs.

**GEOMETRIC REPRESENTATION**

The proofs of the theorems in this section are geometrically-based proofs, in that they use a method of representing the search space in one quadrant of the coordinate plane. Each node \( n \) has a cost \( g(n) \) and a heuristic \( h(n) \) associated with it (for uniform cost search \( h = 0 \)). Thus we can represent the node \( n \) at the point \((i,j)\), where \( i = h(n) \) and \( j = g(n) \). This representation maps the entire search space into one quadrant of the plane (many nodes can be mapped into the same point, but this is of no consequence), and gives us the basis for geometrically describing the search process.

Figure 2 provides an example of the geometric representation of a search space. Note that the node \( n \) is represented at the point \((h(n), g(n))\), and that the \( h \)-axis extends horizontally to the right while the \( g \)-axis extends vertically downward. Since the \( g \)-component of any start node is zero, all start nodes lie on the \( h \)-axis. Similarly, all goal nodes lie on the \( g \)-axis since their \( h \)-components are zero. The nodes \( n_6 \) and \( n_7 \) could be placed in more than one position in the plane since there is more than one path from \( n_6 \) to each. In Figure 2 we have chosen the position which corresponds to the shortest path.

The heuristic used in Figure 2 satisfies the lower bound condition since it can be verified that the \( h(n_i) \) is less than or equal to the distance between \( n_i \) and \( n_g \) for \( i=1, \ldots ,7 \). However, this heuristic is not con-
sistent since (for example) $h(n_1) - h(n_4) = 5 - 2 = 3$, while $k(n_1, n_4) = 2$.

Each choice of $\omega$ in $f_\omega = (1 - \omega)g + \omega h$ determines the direction in which the space will be searched. Search with $\omega = \frac{1}{2}$ is called diagonal search because it defines all nodes which lie on the same diagonal to have equal merit and attempts to expand nodes in the direction indicated in Figure 3(a). Upwards diagonal search differs from diagonal in that node $n$ has better merit than a node $n'$ iff $f(n) < f(n')$, or $h(n) < h(n')$ when $f(n) = f(n')$; and attempts to expand nodes in the same direction as diagonal search except that the search proceeds up successive diagonals as indicated in Figure 3(b). Figures 3(c)-3(f) illustrate the direction of search for various values of $\omega$. If the distinction between the $h$-components is made for nodes with the same $f$-value as is done in upwards diagonal search, then along each line in these figures the search proceeds in the direction of the $h$-axis.

As the process of searching the space proceeds, the region of the quadrant which has been covered grows. As any stage of the search process all nodes which lie inside the covered region (including the boundary) can potentially be expanded. A node $n$ lying inside such a region will be expanded if there is a path from the start node to $n$ such that all the nodes on this path also lie within the region. Note that $n$ lying within the region covered by the search is not sufficient reason for it to be expanded. Note also that a node which is expanded does not necessarily lie on the diagonal of the triangular region, but may lie strictly within the region.

An alternative to specifying an evaluation function in the form $f_\omega = (1 - \omega)g + \omega h$ for $\omega \in [0,1]$ is to use the form $f_\alpha = g + \alpha h$ for $\alpha \in [0, \infty)$. Since scaling the evaluation function does not change the merit ordering defined by it, the relationship between these two forms is given by $\alpha = \omega / (1 - \omega)$. For example, diagonal search is specified by setting either $\omega = \frac{1}{2}$ or $\alpha = 1$ in these respective formulas.

When the first form is used, changes in the value of $\omega$ correspond most naturally to changing the direction of search as illustrated in Figure 3. However, the parameter $\alpha$ in the second form can be thought of as being part of the heuristic component of the evaluation function. When this is done, it is most natural to think in terms of the direction of search remaining diagonal, but all the nodes being moved either towards the $g$-axis if $\alpha < 1$, or away from the $g$-axis if $\alpha > 1$. Thus the above two forms point out the two ways in
which a change in an evaluation function can be viewed; either as a change in the direction of search or as a change in the position of the nodes. In this paper we will use the \( f_\omega = (1-\omega)g + \omega h \) alternative.

**THEOREMS**

Now that we have explained the geometric representation of the search space, we turn to the statements and proofs of the theorems. The proofs of the theorems refer to the illustrations in Figure 4. The term search strategy will mean an ordered search strategy which uses the evaluation function \( f_\omega = (1-\omega)g + \omega h \).

**THEOREM 1. [Completeness]** A search strategy is complete for all \( \delta \)-graphs iff \( \omega \in [0,1) \).

**PROOF:** Assume that there is a solution, that is, a path from the start node \( s \) to a goal node \( t \), and that \( \omega \in (0,1) \). Thus, the direction of search is not parallel to the \( g \)-axis (the case where \( \omega = 1 \)). This case is illustrated in Figure 4(a). Suppose that the solution is not found. Then the search strategy never reached all of the nodes on the solution path because some region in the plane encountered by the search strategy contains an infinite number of nodes. Since \( \omega \neq 1 \), such a region must be either an infinite strip parallel to the \( h \)-axis (the case where \( \omega = 0 \)), or a finite triangular region (the case where \( 0 < \omega < 1 \)). In either case, since some region contains an infinite number of nodes, there exists a path in the graph whose \( g \)-values are unbounded. A path of unbounded \( g \)-values is the only condition which can give rise to an infinite number of nodes in either a region parallel to the \( h \)-axis or a finite triangular region. However, a path of unbounded \( g \)-values is impossible because the graph is \( \delta \)-finite. Hence, if \( \omega \in (0,1) \) then all the nodes along the path from \( s \) to \( t \) will eventually be expanded; that is the search strategy is complete.

Now suppose that \( \omega = 1 \). Then since there can be an infinite path in the graph whose \( h \)-values are bounded above, one of the infinite strips which are encountered by the search strategy before it covers all of the nodes from \( s \) to \( t \) can contain infinitely many nodes. Such a strip would prevent the search strategy from finding the solution. Thus search with \( \omega = 1 \) is incomplete.

---

**Figure 4—Illustrations of different search strategies with various conditions on the heuristic.** Although not explicitly shown in the figures, all nodes which lie in a region of search and have ancestors forming a path back to the start node which also lie in the region of search are expanded.
The admissibility theorem tells us the conditions under which the first solution found is a minimum cost solution.

**THEOREM 2. [Admissibility]** Let \( h \) satisfy the lower bound condition, \( h \leq h_p \), where \( h_p \) is the perfect heuristic. Then search with \( h \) is admissible for \( \delta \)-graphs if \( w \in [0, \frac{1}{2}] \).

**PROOF:** Let us begin by thinking in terms of the nodes positioned in the plane as defined by the perfect heuristic \( h_p \). Since \( g(n) + h_p(n) \) equals the minimum solution cost for each node \( n \), which is on a minimum solution path \( s = n_0, \ldots, n_k = t \), all such \( n \) must lie on the minimum cost diagonal of the plane (see Figure 4(b)). Since \( g(n') + h_p(n') \) is greater than the minimum solution cost for each node \( n' \) which does not lie on the minimum cost diagonal path, all such \( n' \) lie outside the triangular region defined by the maximum cost diagonal. Thus, no nodes lie within this triangular region. If the heuristic is not perfect, but does satisfy the lower bound condition, then the nodes \( n_0, \ldots, n_k \) are all pulled to the left and lie in the triangular region (some may lie on the boundary). Thus, diagonal, upper diagonal, and any search with \( w \geq \frac{1}{2} \) (see Figure 4(c)) will have expanded \( n_0, \ldots, n_{k-1} \) by the time search reaches the location of the minimum goal node \( t \), and therefore \( t \) will be found before any other goal node. Hence if \( w \in [0, \frac{1}{2}] \) the search is admissible.

However, if \( w > \frac{1}{2} \) (see Figure 4(d)) the region covered by the search can include all of the nodes on a nonminimum path like \( s = n_0', \ldots, n_{k'} = t' \) (and thus find a nonminimum solution), before it covers all of the nodes on the minimum path \( s = n_0, \ldots, n_k \). In Figure 4(a) the nodes along the subpath (of the minimum solution path) labeled \( P \) lie outside the region which includes all of \( s = n_0, \ldots, n_k \). These nodes (and all of the successors along the minimum path) will not be expanded, and the minimum goal node will not be found. Thus in \( w \in [0, \frac{1}{2}] \) the search is not admissible.

The optimality theorem tells us that for all of the values of \( w \) which give admissible searches, the use of better heuristics will result in improved searches.

**THEOREM 3. [Optimality]** Let \( w \in [0, \frac{1}{2}] \), and \( h < h_p \) \( \leq h_p \). Then search with \( h \) expands every node expanded by search with \( h_p \) for all \( \delta \)-graphs that contain a minimum solution.

**PROOF:** Again let us begin with the nodes positioned in the plane for the perfect heuristic, where nodes on a minimum solution path lie on the minimum cost diagonal and all others lie outside the triangular region. Now think in terms of the movement of the nodes in the plane which results from using the heuristics \( h \) and \( h_p \) instead of \( h_p \). Since \( h < h_p \), \( h \) pulls all of the nodes into the triangular region which are pulled there by \( h_p \), and others besides. Not all of the nodes which lie inside the triangular region will be expanded. Only those for which there exists a path back to a start node such that all of the nodes on this path also lie in the triangular region will be expanded. However, all of the nodes pulled in by \( h \) that are expanded by \( h_p \) will also be expanded by \( h_p \), because those paths back to a start node whose nodes are placed in the triangular region by \( h_p \) will also be placed there by \( h_p \). Thus search with \( h_p \) expands all of the nodes expanded by search with \( h_p \) and diagonal search (i.e., \( w = \frac{1}{2} \)) is optimal.

For \( w > \frac{1}{2} \) the reason for optimality is similar. The poorer heuristic pulls more nodes inside the triangular region than the better heuristic, only now the triangular region in question is as shown in Figure 4(c).

The completeness and admissibility theorems were stated as iff conditions in terms of the subinterval of \( [0,1] \) in which the properties held. This is not done for the optimality theorem because optimality is defined for admissible heuristics, and admissibility holds only in the subinterval \( [0,\frac{1}{2}] \). Thus since \( w \in [\frac{1}{2},1] \) eliminates admissibility, optimality in this subinterval is not a question.

If the hypothesis of the optimality theorem is changed from \( h < h_p \leq h \) to \( h_p \leq h \), then it is possible that search with \( h_p \) may not expand a node \( n \) which lies on the minimum cost diagonal which is expanded by search with \( h \). This is because a search strategy resolves ties arbitrarily, and search with the poorer heuristic \( h_p \) may choose to expand a node \( n \) which is tied for merit with a minimal goal node, while search with the better heuristic \( h \) chooses not to expand \( n \). Thus, if \( h_p \leq h \) then search with \( h_p \) expands all the nodes expanded by search with \( h \) except possibly for a set of nodes which have the same merit as a minimum goal node.

Let us now turn to the consistency property of a heuristic function. It has been pointed out that in general a search strategy does not expand the nodes of the graph according to the merit ordering. Thus, in general, diagonal search does not expand a node as soon as it falls within the region which has been searched. It must continually backtrack to earlier diagonals to expand nodes. Consistency of the heuristic is a sufficient condition to prevent this from happening.

**THEOREM 4.** If \( h \) is consistent then diagonal search (i.e., \( w = \frac{1}{2} \)) never has to backtrack to an earlier diagonal to expand a node.

**PROOF:** If \( h \) is consistent, then if \( k(n,n') \) exists then \( h(n) - h(n') \leq k(n,n') \). What does this condition mean in terms of the coordinate representation of the search space?

\[
\begin{align*}
h(n) - h(n') &\leq g(n') - g(n) \\
\iff g(n) + h(n) &\leq g(n') + h(n') \\
\iff h(n) &\leq k(n,n') \\
\iff f(n) &\leq f(n')
\end{align*}
\]

Thus consistency means that if \( n \) precedes \( n' \) in the graph (which is a necessary and sufficient condition for \( k(n,n') \) to exist), then \( f(n) \leq f(n') \). In terms of the
coordinate representation this means that \( n \) appears on
the same or an earlier diagonal. Hence diagonal search
moves from diagonal to diagonal to expand nodes, and
never has to backtrack to an earlier diagonal to expand
a node (see Figure 4 (f)).

An important consequence of Theorem 4 is that if a
heuristic is consistent then an ordered search strategy
will never have to expand a node more than once; the
first path found to any node is the minimum cost path
to that node. Thus, when the heuristic is consistent,
Case 3 (in the algorithm given in Figure 1), where
\( \text{mS} \) and the new merit is better than the old merit,
will not arise.

Sometimes the concept of a monotonic evaluation
function is used instead of consistency. An evaluation
function \( f \) is monotonic if for all \( n \) and \( n' \), \( n \leq n' \Rightarrow f(n)
\leq f(n') \), and \( h(n') = 0 \) when \( n' \) is a goal node; where
\( n \prec n' \) means that \( n \) precedes \( n' \) in the graph, and \( n \leq n' \)
means that either \( n \) precedes \( n' \) or \( n \) equals \( n' \). For
\( w = \frac{1}{2} \) the concept of monotonicity is equivalent to that
of consistency.

At one time it was thought that the optimality property
required the consistency assumption. Recall that
the proof of the optimality theorem depended on the
poorer heuristic \( h_1 \), pulling all of the nodes into the
triangular region which are pulled in by the better
heuristic \( h_2 \). Thus all of the nodes pulled in and
expanded by \( h_2 \) were pulled in and expanded by \( h_1 \). If the
consistency of \( h_1 \) is assumed, then every node pulled
in by \( h_1 \) is also expanded by \( h_2 \). Thus we can say
every node pulled in by \( h_2 \) is pulled in and expanded by \( h_2 \).
However, it is clear that the optimality property does
not depend on \( h_1 \) expanding every node which it pulls
into the triangular region. That is, the consistency as­
sumption is not necessary for optimality.

THEOREM 5. [Perfect Heuristic] Suppose that all
arcs of the graph have unit costs. Then search with
\( f_0 = (1 - w)g + wh \) only expands nodes on a minimum
solution path iff \( \{ \frac{1}{2}, 1 \} \).

PROOF: As indicated earlier, a perfect heuristic places
no nodes inside the triangular region defined by
the minimum cost diagonal, and places all nodes on
all minimum solution paths on the minimum cost di­
gonal itself. No nodes will be expanded until the area
covered by the search includes the start node (see
Figure 4 (f)). To prove the theorem we must show
that at each stage of the search the node that is ex­
panded lies on the minimum cost diagonal.

Let \( s = n_0, n_1, \ldots, n_t = t \) be a minimum solution path.
Clearly the first stage expands a node on the minimum
cost diagonal since the node expanded is \( n_0 \). Suppose
that \( n_0, n_1, \ldots, n_t \) are the only nodes which have been
expanded at the \( t \)th stage. Then since all arcs have
unit arc costs the successor of \( n_t \) which is on the mini­
mum solution path namely \( n_{t+1} \), lies on the diagonal
and a successor \( n' \) which is not on the minimum solu­
tion path lies off of the diagonal to the right of \( n_{t+1} \) (see
Figure 4 (f)). Thus \( n_{t+1} \), has better merit than the other
successors of \( n_t \). \( n_{t+1} \) has better merit than any previ­
ously unexpanded successor \( n'' \), because \( n_t \) has better
merit than \( n'' \), and \( n_t \), has better merit than \( n_t \). Hence,
\( n_{t+1} \), will be expanded at stage \( t+1 \), and only nodes on
the minimum solutions path are expanded.

If \( w = \{ \frac{1}{2}, 1 \} \) then it is possible for the previously un­
expanded successor \( n' \) to have better merit than \( n_{t+1} \).
Hence a node not on the minimum solution path may be
expanded.

If the unit arc costs assumption is removed from the
hypothesis of Theorem 5 the conclusion is not true.
This is because the node \( n' \) may be in the region which
gives it a better merit than \( n_{t+1} \), and a node off of the
minimum solution path will be expanded. It is clear
that the unit arc cost assumption can be replaced by a
constant arc cost assumption. Figure 5 illustrates
that Theorem 5 does not generalize to arbitrary arc

All values of \( 0 \leq w \leq \frac{1}{2} \) are equally good in the sense
that completeness, admissibility, and optimality all
hold for \( w = \{ \frac{1}{2}, 1 \} \) provided \( h \) satisfies the lowerbound
condition. However, it is clear from an examination
of the regions expanded by searching with various
values of \( w = \{ \frac{1}{2}, 1 \} \) that \( w = \frac{1}{2} \) is best, because it expands
fewest nodes.

Ignoring admissibility and optimality for the mo­

![Figure 5](https://via.placeholder.com/150)

Figure 5—An illustration that Theorem 5 does not generalize
to arbitrary arc costs. Searching the above graph with the
perfect heuristic using \( w = \frac{2}{3} \) not only expand the	node \( n' \) which is off of the minimum solution path,
it even finds the nonminimal solution \( n_{t+1} \), \( n' \), \( n_0 \)
ment, there seem to be intuitive reasons for choosing \( w = 1 \), or at least for choosing \( w > \frac{1}{2} \). Recall that \( g \) is the cost-to-date component of the evaluation function, and that \( h \) estimates the cost to the nearest goal. An intuitive reason for choosing \( w = 1 \) is as follows. Since the search has progressed to the point of generating \( n \), why should we concern ourselves with the cost that has been incurred to get there? Perhaps we should only concern ourselves with what is the expected cost to get to a goal.

On the other hand, in addition to the fact that we may desire a minimum solution, which is not guaranteed when \( w > \frac{1}{2} \), there is the following reason for including a \( g \)-component in the evaluation function. Sometimes it is difficult to construct a good heuristic function for a problem, and in these cases the presence of a \( g \)-component does not hinder the search, and sometimes helps it. Consider, for example, a heuristic function \( h \) which is of bounded error (\( h(n) - \epsilon \leq h(n) \leq h(n) + \epsilon \), for some \( \epsilon > 0 \) and all \( n \)), and which is defined in a way that deliberately misleads the search by being as optimistic as possible for nodes off the minimum solution path and as pessimistic as possible for nodes on the minimum solution path. It has been shown that for such a heuristic, diagonal search \( (w = \frac{1}{2}) \) expands fewer nodes in obtaining a solution than does pure heuristic search \((w = 1)\). This is an extreme example, but it does illustrate how a \( g \)-component can act as a stabilizing source when a poor heuristic is used.

**BIBLIOGRAPHICAL REMARKS AND CONCLUSION**

Graph representations have long been recognized as important models in problem solving. Discussion of both early and more recent use of graph representations can be found in the books by Ernst and Newell, Barnerji, Nilsson, and Slagle. These books along with the articles by Amarel, Michie, and Sandewall also contain general discussions of ordered search strategies.

The admissibility and optimality of ordered search strategies and the concepts related to these theorems are due to Hart, Nilsson, and Raphael. Algebraic proofs of these theorems also appear in Nilsson. In this reference a \( \delta \)-graph is defined to be a graph whose arc costs are bounded away from zero. The definition used in this paper which restricts the partial sums of the arc costs along an infinite path from being unbounded is more general, since it allows finitely many arcs of zero cost. This definition is based on the concept of a \( \delta \)-finite merit ordering which was used by Kowalski, who also was the first to use the geometric representation. Pohl investigated the evaluation function \( f_w = (1-w)g + wh \) in some experiments on the 15-puzzle. Doran and Michie also experimented with the 15-puzzle. The theorem on completeness was proved by Pohl. The theorem on the perfect heuristic and the argument for including a \( g \)-component in evaluation function are also due to Pohl. To the best of our knowledge, the fact that the theorem on the perfect heuristic does not generalize to arbitrary arc costs has not previously been pointed out.

Hart, Nilsson, and Raphael developed ordered search strategies for directed graphs, which are used to represent state-space representations. Chang and Slagle developed ordered search strategies for AND/OR graphs, which are used for problem-reduction representations. Ordered search strategies for theorem-proving graphs, which are used to represent state-space representations with multiple-input operators, were developed by Kowalski, and also by Michie and Siebert. An illustration of how the direction in which a problem space is searched determines whether a single problem representation is viewed either as a state-space or a problem-reduction representation appears in VanderBrug and Minker.

The formal properties of ordered search algorithms, which we proved using the geometric representation, are of limited interest to the developers of a practical problem solving system. Completeness is rarely a big consideration in the design of such a system. Usually what keeps a search from being successful is exhaustion of the available resources, not the incompleteness of the search strategy. Admissibility is only sometimes an important consideration, since often a solution which is approximately minimum is sufficient. Thus, one can use a heuristic which only approximately satisfies the lowerbound condition \( h \leq h_p + \epsilon \), and expect to get a solution which is approximately minimum. Such a search is less conservative (in the sense that it takes more chances) than an admissible one, and frequently will find an approximately minimum solution before the admissible search finds a minimum one.

Nonetheless, the theorems are important formal properties, and help to unify the work done in this area. We believe that the geometric approach to the presentation of these formal properties is an intuitive alternative to the algebraic approach.

**ACKNOWLEDGMENTS**

The support of the National Aeronautics and Space Administration under Grant NGR-21-002-270, and the assistance of L. Kanal, J. Minker and A. Rosenfeld is gratefully acknowledged.

**REFERENCES**

2. Pohl, I., "Heuristic Search Viewed as Path Finding in a


