ADVEI2

ELSEVIER

ADVANCED ENGINEERING INFORMATICS

Advanced Engineering Informatics 00 (2002) 000–000

www.elsevier.com/locate/aei

A performance-based approach to wheelchair accessible route analysis

Charles S. Han[1], Kincho H. Law*, Jean-Claude Latombe[2], John C. Kunz

[a]*Center for Integrated Facility Engineering, Stanford University, Stanford, CA 94305-4020, USA*

Received 7 September 2001; revised 29 October 2001; accepted 2 November 2001

Abstract

This paper presents a method to determine if a usable wheelchair accessible route in a facility exists using motion-planning techniques. We use a ‘performance-based’ approach to predict the performance of a facility design against requirements of a building code. This approach has advantages over the traditional ‘prescriptive’ code-based approach for assessing acceptability of designs, which is normal practice today for assessing wheelchair accessibility. The prescriptive method can be ambiguous, contradictory, complex, and unduly restrictive in practice, and it can be ad hoc and difficult to implement as a computer application. The performance-based approach directly models the actual possible behaviors of an artifact (in this case, wheelchair motion) that are related to the functional intent of the designed system (a building) and (hopefully) to the specification of a prescriptive building code. This paper presents example cases from architectural practice to illustrate the use of robot motion-planning techniques for wheelchair accessibility analysis. This application is an example of using modern computational methods in support of knowledge-intensive engineering. The simulation method has broad applicability within engineering design. We illustrate and discuss how to analyze virtual simulations of the detailed behavior of a designed artifact in order to assess its use by intended users.

Keywords: Motion planning; Disabled access; Wheelchair accessibility; Performance-based analysis

1. Introduction

This paper develops a method to determine if a usable wheelchair accessible route in a facility exists using computer-based motion-planning techniques. One concern for designing a facility is the extent to which it satisfies a set of usability objectives. In the US, wheelchair access in private facilities is often an important objective, and certain wheelchair accessibility is a constraint that is mandated by law for most public facilities. The *Americans with Disabilities Act Accessibility Guidelines* (ADAAG) contain ‘prescriptive’ specifications for determining the existence of a valid wheelchair accessible route as well as other objectives for disabled access

Advantages of using prescriptive provisions include straightforward evaluation of a design using the prescribed parameters, and such evaluation often does not need high-level engineering knowledge about the specific analysis. However, prescriptive-based codes can be ambiguous, contradictory, complex, and restrictive [6]. Solutions constrained by prescriptive-based codes such as the ADAAG address only a fraction of the possible solutions that meet the design intent or objectives of these codes. Since it often is implicit, it is often difficult for both designers and code checkers to discern the design intent and objectives of a building code or code provision. However, in the case of the ADAAG, the intent is clearly stated as “…scoping and technical requirements for accessibility to buildings and facilities by individuals with disabilities…”

Furthermore, instances exist in which adhering to these prescriptive provisions produces a design that may not be usable.

As a partial solution to the problems of prescriptive-based building codes, many jurisdictions have adopted or are moving toward the adoption of ‘performance-based’ codes. We use the term ‘performance-based’ to imply the performance computed by simulating behavior of models (in this case, a wheelchair in the configuration of a facility).

For example, California provides a performance-based alternative to its prescriptive-based energy codes [2]. As opposed to prescriptive-based codes that provide solutions

* Address: Department of Civil and Environmental Engineering, Stanford University, Stanford, CA 94305-4020, USA.

E-mail addresses: chuck.han@autodesk.com (C.S. Han), law@ce.stanford.edu (K.H. Law), latombe@cs.stanford.edu (J.-C. Latombe), kunz@stanford.edu (J.C. Kunz).

[1] Autodesk, Inc., San Rafael, CA 94903, USA.

[2] Department of Computer Science, Stanford University, Stanford, CA 94305, USA.

1474-0346/02/$ - see front matter
PII: S1474-0346(01)00003-9

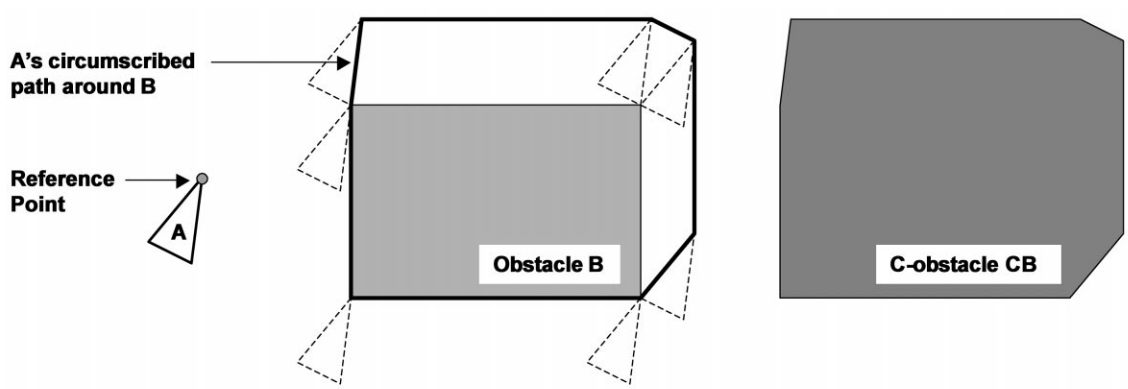

Fig. 1. Generation of configuration space. A robot (A) and an obstacle (B) exist in a workspace. The motion planner grows the obstacle in configuration space *C* by adding to B the shadow of A as it moves about the perimeter of B. The planner then reduces the robot to a reference point.

abstracted from the design intent or objective of a building code, performance-based codes attempt to directly capture the behaviors that conform to the intent of the design codes or regulations. This direct performance-based approach accepts design solutions that satisfy the usability constraints, including those solutions that do not comply with the prescriptive-based constraints specified by a design code. When a performance-based approach accurately models usability, this approach will identify and reject design solutions that are not usable—some unusable designs may be accepted using the prescriptive-based constraints specified by a design code.

This paper presents the methods developed for accessible route analysis using motion-planning simulation to capture the behavior of a moving wheelchair. First, we give a brief overview of motion planning and the technique adopted in this work. We then provide a definition of an accessible route and its components. Methodologies for the analysis of accessible components are then discussed. Application examples are provided to demonstrate the benefits of the performance-based approach. Application of the approach for the analysis of a floor plan is also given. Finally, this paper concludes with a short summary and discussion for future considerations.

2. Basics of motion planning

In basic motion-planning, a robot A moves through a Euclidean space *W* (the *workspace*) represented as R^N where *R* is the set of real numbers, and $N = 2$ or 3 is the spatial dimension. The motion planner for the wheelchair assumes a two-dimensional space where $N = 2$. The space *W* is populated with obstacles represented as $B_1, B_2, \ldots, B_q$, and the motion planner parameters are defined by the shape, position, and orientation of A, the B_is, and W. Given the initial and goal positions and their orientations, the objective of the motion planner is to determine if a path exists from the initial to the goal positions and, if so, to generate a continuous path τ through the workspace *W* for the robot A avoiding the obstacles B_is. To find a path in a space, the method first approximates the wheelchair by a disc. Then, it grows the obstacles isotropically by the radius of this disc. Finally, the motion planner computes paths between given points of the resulting free space. The remainder of this section elaborates the way we applied the approach for the wheelchair problem.

The motion planner generates a *configuration space C* from the geometric properties of A, the B_is, and *W*, and it attempts to construct a path in this configuration space. In the new space C, the motion planner transforms robot A to a point object, and the motion-planning problem becomes one of generating the path τ in C. For a 2-dimensional space W, the dimension *m* of the configuration space *C* is 3. For example, a robot A restricted to move in the *xy*-plane ($W = R^2$) has three degrees of freedom: translations in the *x* and *y* directions and the orientation θ. Working in the configuration space *C* instead of the workspace *W*, the constraints become more explicit. If the motion planner works directly with the workspace *W*, it would have to perform operations such as collision checking at each proposed path position in C, the collision-checking operation has already been addressed for all possible robot positions.

As the motion-planner maps or 'shrinks' A to a point object, an obstacle B_i maps to the C-obstacle CB_i by 'growing' its shape based on the geometric parameters of A and B_i as shown in Fig. 1. The basic algorithm establishes a reference point with respect to the robot A and tracing A around the obstacle B_i. The path circumscribed by A describes the C-obstacle CB_i. If A can freely rotate, the shape of CB_i depends on A's orientation. Fig. 1 illustrates the transformation of an obstacle to a C-obstacle, given a fixed orientation of A.

By generating the configuration space C, the motion planner transforms the path-planning problem into the problem of finding a smooth curve within C. Now, the motion planner must guide the robot from the initial point to the goal point through C. Latombe [7] notes that using some type of 'potential field' is the most successful method for guiding the robot A. The generated potential field guides

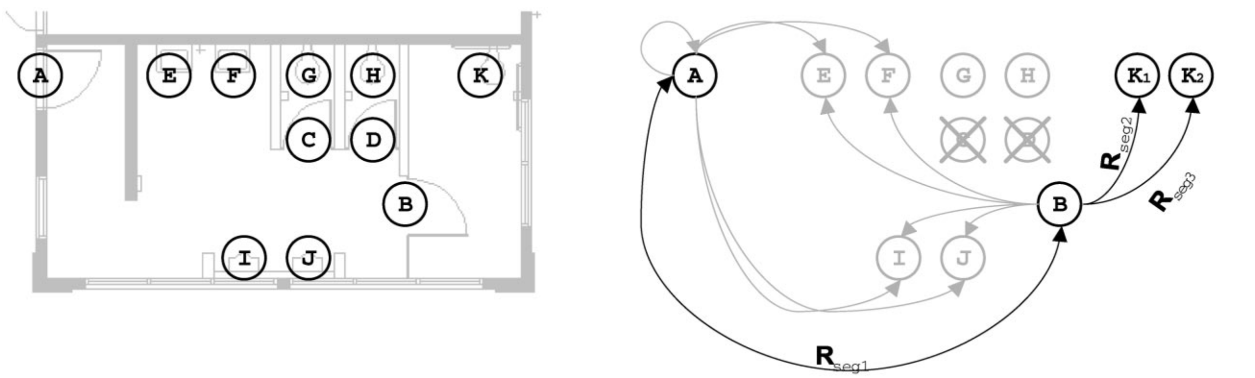

Fig. 2. Potential accessible route components and accessible graph of a bathroom facility. The left figure shows the components (A–K) of a multi-occupant bathroom. The bold nodes and arcs in the access graph on the right show the path from the entry door to the accessible toilet. The graph shows one accessible route segment R_{seg1} from A to B and two accessible route segments from B to K. Note that the path planner found that G and H are inaccessible from A.

A by forcing it down the gradient from the initial point to the goal point. The motion planner discretizes C by defining a grid over the space and generates the potential field values for each grid cell.

Since the motion planner knows the geometric parameters of A, the B_is, and W a priori, it can generate potential fields free of local minima. The accessible route analysis developed in this paper uses a potential-field-generating algorithm NF1 as described by Ref. [7], which can be shown to be free of local minima. The algorithm creates a potential field that guides the robot A from the initial point to the goal point on a path τ that grazes the C-obstacles.

3. Definition of the accessible route

The ADAAG defines an accessible route as:

> *Provision 3.5 Definitions. Accessible Route*. A continuous unobstructed path connecting all accessible elements and spaces of a building or facility…

In addition to the above definition, the ADAAG prescribes measurements that define accessibility for various building elements (such as doors and toilets) along an accessible route. An accessible route can thus be described as a sequence of accessible route segments and, if needed, adequate clearance at the openings of critical components of a space.

Our motion planning technique determines accessible route segments in a space between building elements (such as doors and toilets). In addition, individual elements are checked for geometric clearances. A complete accessible route includes many accessible components (including openings and route segments). The route is considered accessible if all components in the route are accessible.

The following definitions define the terms that we use in the performance based accessibility analysis:

R_{init} the initial position (the starting point of an accessible route or segment of accessible route)

R_{goal} the goal position (the ending point of an accessible route or segment of accessible route)

R_{seg} a segment of the accessible route between the initial position R_{init} and the final position R_{goal} within a space

R_{open} the clearance area at an opening of building elements

The motion planner generates a path between an *initial* point and a *goal* point. Building components along the accessible route graph map to the R nodes: R_{open} nodes map to initial and goal points, R_{init} nodes map to initial points, and R_{goal} nodes map to goal points. The arcs of the graph (the R_{seg} components) map to the generated path between the R nodes. Fig. 2 shows the potential accessible route segments and components of a bathroom facility and the associated accessible graph. Route segments (arcs) are established between adjacent building elements. It is interesting to note that there are two established goal nodes (and two route segments) for the toilet at K because the algorithm models access to the toilet using either a side or a diagonal transfer. Furthermore, nodes A and B are potential accessible openings but the doorways at C and D are eliminated as potential R_{open} node since they do not have sufficient clearance requirements. Also note that each opening R_{open} and each route segment R_{seg} can be evaluated independently.

In the following, we first discuss the application of a motion planner for determining the existence of an accessible route segment R_{seg}. We then discuss the compliance checking of the component openings, R_{open}. Finally, the determination of the initial and goal positions of various building elements is discussed.

4. The wheelchair motion planner and determination of an accessible route segment R_{seg}

Although the goal of this study is to directly model the motion of a wheelchair through a space using the actual wheelchair geometry, the motion planner first tries to verify

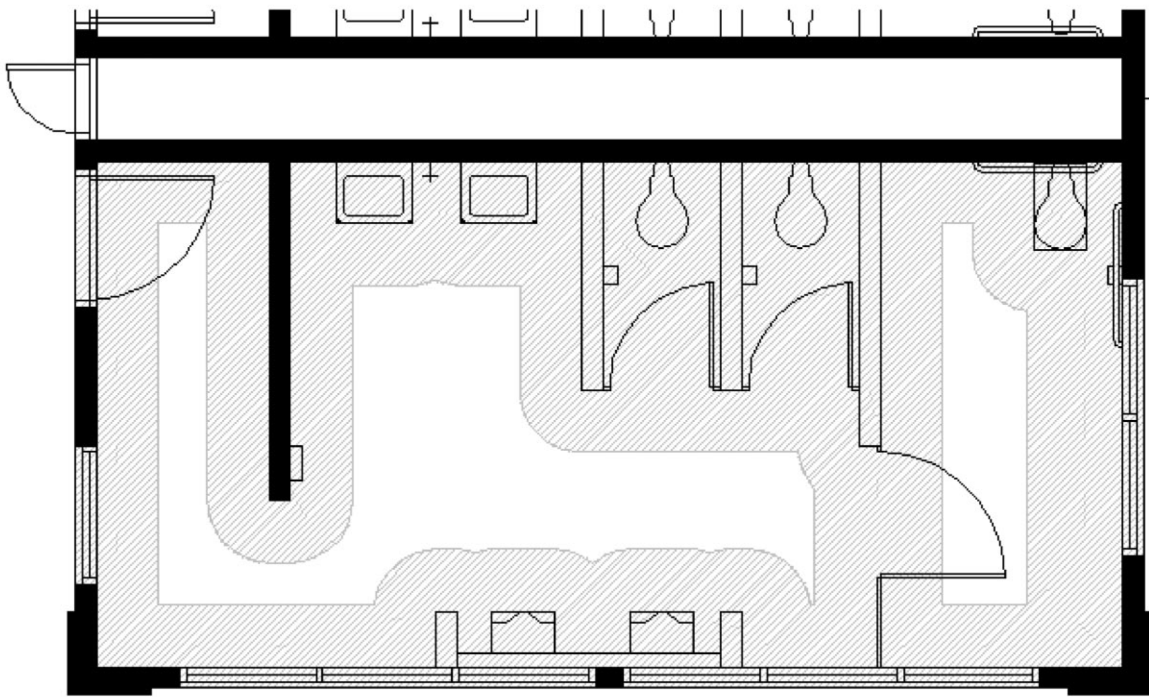

Fig. 3. The C_{36} configuration space for a bathroom facility. White areas show spaces in which a 36 in. disk can move.

the existence of a 'comfortable' width along the route. To determine an accessible route segment within a space, the motion planner performs two basic tasks:

1. Verify the existence of adequate clearance width along the route.
2. Determine if a wheelchair user can negotiate this route given the assumed geometric and behavioral constraints of the wheelchair.

4.1. Pass one: determining the clearance width of the accessible route

The existence of an accessible route is intended to ensure the usability of a facility for wheelchair-bound users, and in most cases, the wheelchair user should be able to negotiate the accessible route using only forward motion. The ADAAG provision below prescribes the width parameters of the route.

> *Provision 4.3.3 Width*. The minimum clear width of an accessible route shall be 36 in. (915 mm) except at doors (see 4.13.5 and 4.13.6). If a person in a wheelchair must make a turn around an obstruction, the minimum clear width of the accessible route shall be as shown in Fig. 7(a) and (b).[3]

For the first pass, we focus on the general 36 in. (915 mm) requirement on the accessible route. The exception rule for turn around will be discussed later for constructing the second pass. The general 36 in. (915 mm) width rule does not represent the width of a wheelchair but prescribes a 'comfortable' width for the wheelchair user to negotiate. To satisfy the provision, the motion planner uses a 36 in. (915 mm) disc to describe the geometry of the robot A_{36}. Given the workspace W as determined by the floor space and the building components, the motion planner generates the configuration space C_{36} given A_{36} and W. The path generated by this planner is not subject to geometric and physical constraints (such as turning radius) of a robot—the robot simply slips and slides down the potential gradient. This type of planner is known as a holonomic planner [7]. Fig. 3 illustrates the C_{36} configuration space for a bathroom facility example. The white (non-shaped) areas represent legal positions for the A_{36} disc robot and are ensured to provide a 36 in. (915 mm) clearance. Note that the motion planner treats a doorway with the door in the closed position, and, hence, in the configuration space between the entry door and the accessible toilet is discontinuous.

For this pass, the motion planner does not actually generate the path. It simply generates the potential field from the goal points to the initial points. If the potential-field generation cannot reach the initial point, there is no 36 in. (915 mm) width path between the two points, and an accessible wheelchair route between the points does not exist. If, however, the potential-field generation does reach the initial point signifying that a 36 in. (915 mm) wide path does exist between the two points, the motion planner proceeds to the second pass. Fig. 4 illustrates the actual potential field generated between the entry door and a urinal using a

[3] Note that ADAAG Fig. 7(a) and (b) are shown in this paper as Fig. 5.

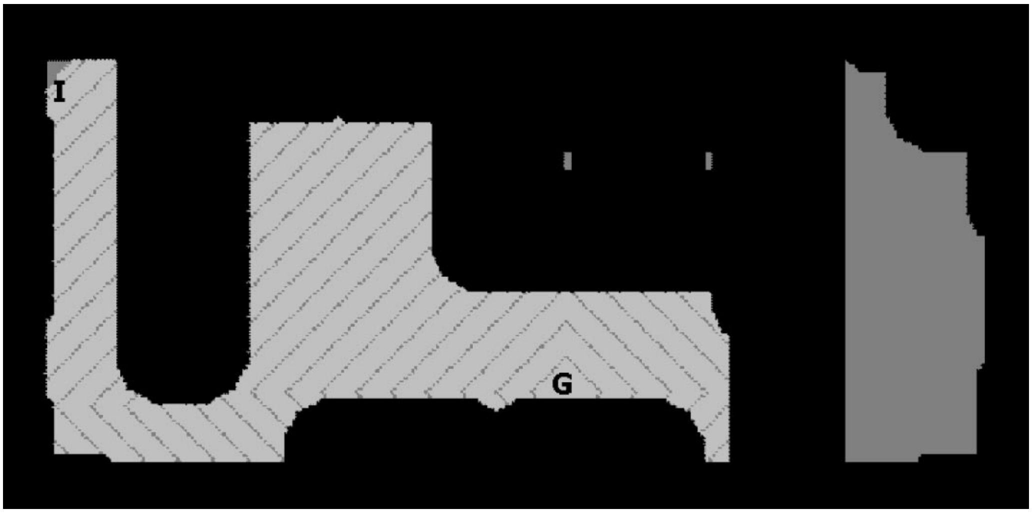

Fig. 4. The potential field between the initial entry door and the goal right urinal of the bathroom design of Fig. 2. The robot path-planner finds the potential field and finds any routes within the field.

1 in. cell discretization and a 12-sided polygon with a 36 in. (915 mm) diameter for the A_{36} robot. The contour lines illustrate the rectangular 'Manhattan' nature of the generated potential field [7].

4.2. Pass two: capturing the characteristics of wheelchair motion

To closely examine exceptional rules, e.g. the clear width of an accessible route around an obstruction, we need to be able to capture the behavior of wheelchair motion. Fig. 5 illustrates the two ADAAG figures (Fig. 7(a) and (b)) referenced by Provision 4.3.3 of the ADAAG. Note that in the prescriptive width definition, the exceptional rules and the associated figures do not address all possible turn-around configurations. In order to provide accessibility check on all possible configurations, the route planner models, as closely as possible, the behavior of wheelchair motion as to the level of detail set by the guidelines.

In the second pass, in addition to the C_{36} configuration

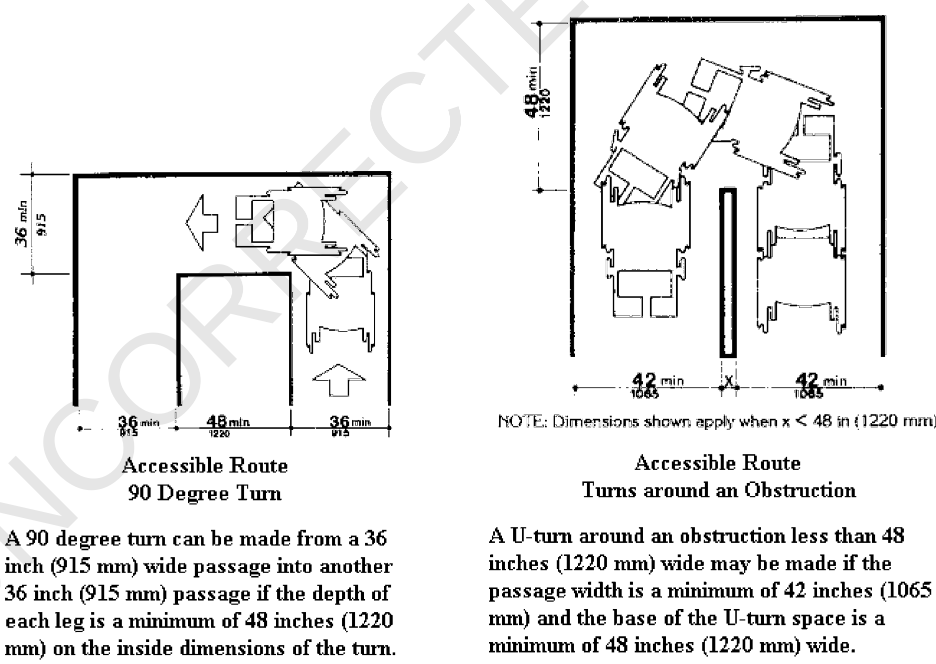

Fig. 5. Minimum accessible route turning clearances defined in Provision 4.3.3 of the ADAAG. These exceptional provisions are examples of the many exceptional rules that prescriptive code specifications represent explicitly.

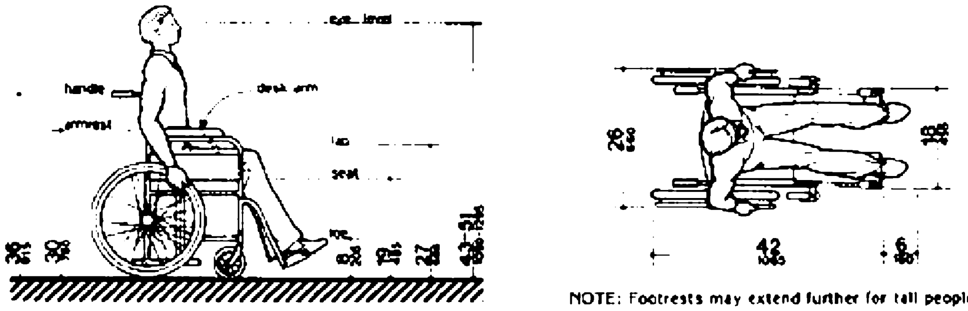

Fig. 6. Wheelchair dimensions as shown in the ADAAG. The wheelchair model represents these design specifications explicitly.

space generated in the first pass, configuration spaces are also generated using the actual wheelchair robot denoted as A_{wc}. Fig. 6 shows the reference wheelchair dimensions as given in the ADAAG, and Fig. 7 illustrates the geometry of the wheelchair robot which has its width less than 36 in. (915 mm) wide. The potential fields within the configuration spaces generated by the planner are used as a guide to determine the path τ for A_{wc}.

Since A_{wc} is not a disc, the motion planner must keep track of the robot's orientation while generating a path, and the motion planner must check each wheelchair position and orientation against the obstacles in the space. The motion planner discretizes the rotation space into n configuration spaces $C_{wc0}\dots C_{wcn}$ where the angle (in radians) between two consecutive orientations A_{wc} is equal to $2\pi/n$. Now, the motion planner can check the wheelchair position and orientation (x, y, θ) against each appropriate configuration space C_{wci}. The motion planner generates a nonholonomic path [8] by restricting A_{wc} to three moves: a left turn, a right turn, and a straight-ahead move. Fig. 8 illustrates these three options. For the left and right turns, the motion planner describes the vertex of the turning angle as the perpendicular length r from the centerpoint between the major wheelchair wheels. The motion planner records the actual position of A_{wc} at the centerpoint of the half-dodecagon at the front of the robot. The displacement distance D from either turn (which is dependent on r) dictates the translation of A_{wc}.

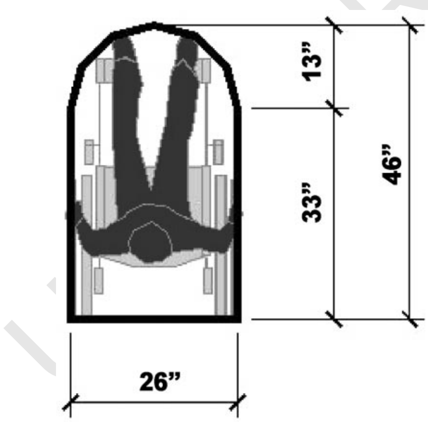

Fig. 7. Dimensions of the robot A_{wc}. The robot path-planner represents the dimensions shown in this figure in its model of a wheelchair.

4.2.1. Determination of turning radii for the wheelchair

The performance-based accessible route path planner employs two values r_1 and r_2 for the turning radius r to allow adjustment for maneuvering the wheelchair. The larger turning radius r_1 is employed to move the wheelchair robot to the goal point. As the wheelchair user nears a goal, the user naturally slows down allowing finer maneuvering with a smaller turning radius r_2. In this study, when the wheelchair has moved within an 18 in. (460 mm) locus of the goal point, the motion planner switches to the smaller turning radius r_2 to try to maneuver A_{wc} to the goal point with an acceptable orientation.

In determining the value for the turning radius, a larger value represents a larger turning circle and a more comfortable path τ for the wheelchair user. The largest possible value for r_1 was determined using a trial and error process using the two turning-around-an-obstruction configurations based the ADAAG Provision 4.3.3 shown in Fig. 5. Figs. 9 and 10 illustrate the first legal paths with an r_1-value that works for both turning-around-an-obstruction configurations produced by a trial-and-error process. Through this process, we selected 24 in. (610 mm) as the value of the larger turning radius r_1.

As with the determination of r_1, a trial-and-error process was employed to determine the smaller turning radius r_2 based on Provision 4.2.3 of the ADAAG:

> *Provision 4.2.3 Wheelchair Turning Space*. The space required for a wheelchair to make a 180° turn is a clear space of 60 in (1525 mm) diameter (see Fig. 3(a)) or a T-shaped space (see Fig. 3(b)).[4]

Fig. 10 shows the ADAAG figures associated with this provision. The turning radius r_2 is determined by finding the maximum turning radius that can make A_{wc} perform the turning maneuver in a 60 in. (1525 mm) space. Fig. 11 illustrates the turning maneuver that satisfies the 60 in. (1525 mm) width constraint (as shown by the dimension

[4] Note that ADAAG Fig. 3(a) and (b) are shown in this paper as Fig. 10.

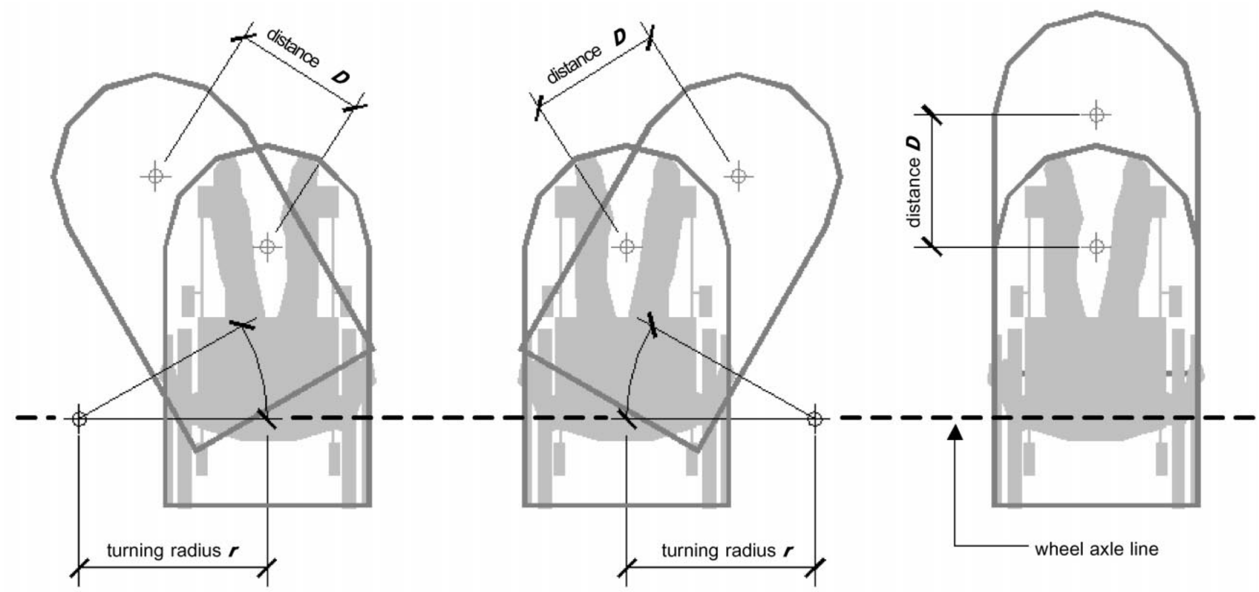

Fig. 8. The three movement behaviors (move left, right, and forward) for the A_{wc} robot.

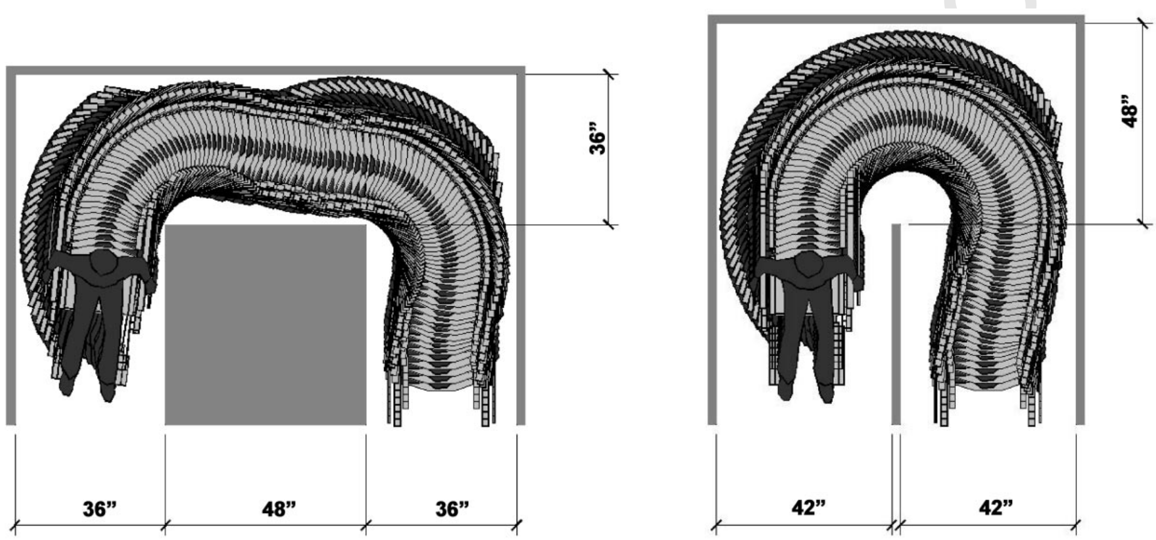

Fig. 9. Determination of the large turning radius ($r_1 = 24''$ (610 mm)) for the exception in Provision 4.3.3 of the ADAAG.

line) using a turning radius r_2 of 9 in. (230 mm). It should be noted that while this clearance width of 60 in. (1525 mm) is sufficient, the necessary clearance width orthogonal to the dimensioned clearance exceeds the 60 in. (1525 mm) diameter requirement. Indeed, this clearance width, in practice, should exceed the 60 in. (1525 mm) as discussed in the Appendix of the ADAAG:

> *Provision A4.2.3 Wheelchair Turning Space*. These guidelines specify a minimum space of 60 in. (1525 mm) diameter or a 60 in. by 60 in. (1525 mm by 1525 mm) T-shaped space for a pivoting 180° turn of a wheelchair. This space is usually satisfactory for turning around, but many people will not be able to turn without repeated tries and bumping into surrounding objects. The space shown in *Fig. A2* will allow most wheelchair users to complete U-turns without difficulty.[5]

Fig. 12 (ADAAG *Fig. A2*) illustrates an acceptable clearance oval, and the turning movement as shown in Fig. 11 fits into the suggested oval geometry.

4.2.2. Wheelchair motion planner and path generation

The motion planner uses the potential field in the configuration space C_{36} to guide the wheelchair robot A_{wc} as follows: starting from the initial position and orientation

[5] Note that ADAAG *Fig. A2* is shown in this paper as Fig. 12.

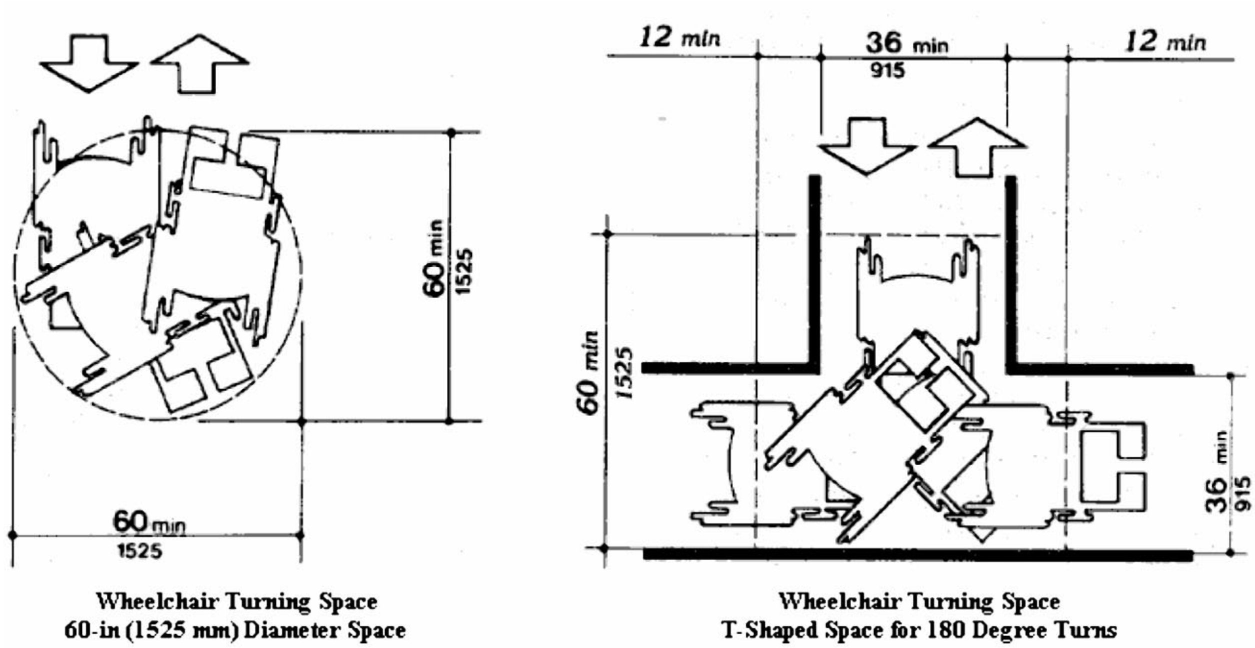

Fig. 10. The prescribed turning circle and T-space from the ADAAG.

q_{init}, the motion planner examines the move options for left, right, and straight ahead (denoted as q_{left}, q_{right}, and $q_{straight}$, respectively) using r_1 as the turning radius for the robot.

1. If q_{left} resides in the C_{36} and appropriate C_{wci} configuration space, the motion planner compares the position (x, y) associated with q_{left} with the position (x, y) associated with q_{goal}.
2. If the two positions are not the same, the motion planner looks up the potential field of the position and inserts the node into a priority queue, which prioritizes the nodes by their potential field value (the lower the value, the higher the priority).
3. Finally, the motion planner inserts a pointer to the previous position (in this case, q_{init}) in the node and marks q_{left} in the appropriate C_{wci} configuration space potential field as having been already visited.

The motion planner repeats this procedure for q_{right} and $q_{straight}$.

The motion planner continues the iterative process by removing the highest priority node (the node with the lowest potential value) from the priority queue and examining the

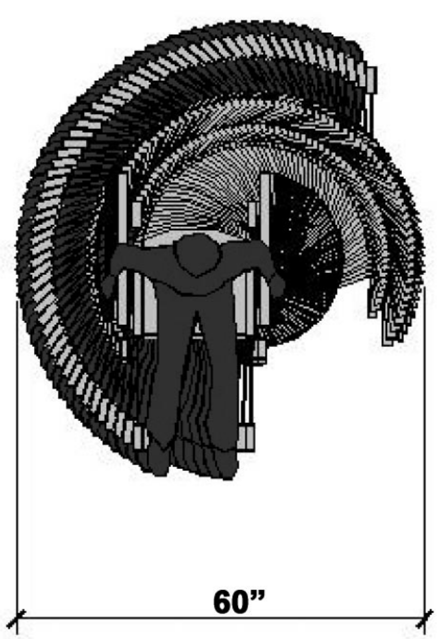

Fig. 11. Determination of the small turning radius ($r_2 = 9''$ (230 mm)) for use by the motion planner to address the issue of Provision 4.2.3 of the ADAAG.

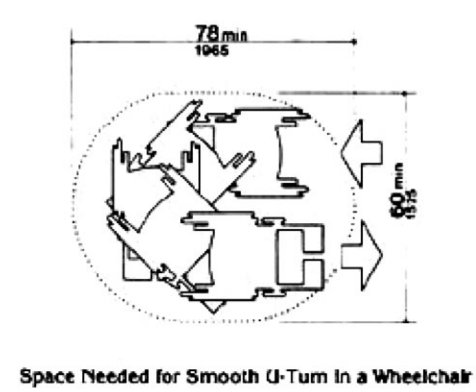

Fig. 12. The ADAAG turning clearance geometry.

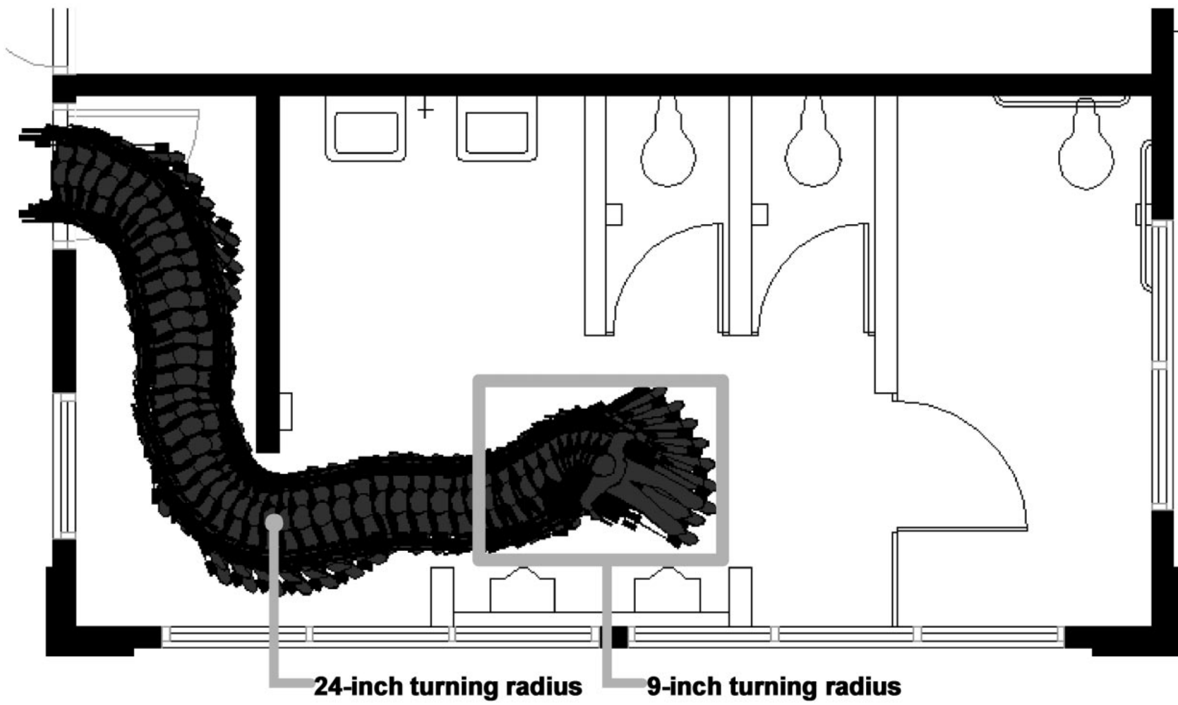

Fig. 13. The generated path from the initial doorway entry to the goal right urinal in the bathroom facility. The first part of the path uses the larger r_1 turning radius, and the last part of the path uses the smaller r_2 turning radius.

three move options from the associated position and orientation q. The C_{wci} configuration spaces include the visited as well as the free space information, and the motion planner treats a visited q_{init} as an obstacle. When q is within an 18 in. (460 mm) locus of q_{goal}, the planner starts generating new positions using the smaller turning radius r_2. When the robot reaches the goal position, the motion planner examines whether the orientation θ associated with the current q is within the allowable range of θ_{goal} and, if so, it records the path τ. The iterative process continues until either the motion planner empties the priority queue (indicating no path τ exists) or the position (x, y) associated with the current q matches with q_{goal} for both position and the acceptable orientation range.Fig. 13 illustrates a generated path from the entry door to the urinal in the bathroom facility.

5. Accessibility analysis of R_{open} components

ADAAG prescribes wheelchair clearances at doors and entrances along an accessible route. The R_{open} node of an accessible route graph consists of three clearance components: the clearance of the opening and clearances on either side of the opening. For the opening, the analysis applies a geometric test with the parameters of the required clearance box taken directly from the following provision:

Provision 4.13.5 Clear Width. Doorways shall have a minimum clear opening of 32 in. (815 mm) with the door open 90°, measured between the face of the door and the opposite stop (see Fig. 24(a)–(d)). Openings more than 24 in. (610 mm) in depth shall comply with Provisions 4.2.1 and 4.3.3 (see Fig. 24(e)).[6]
EXCEPTION: Doors not requiring full user passage, such as shallow closets, may have the clear opening reduced to 20 in. (510 mm) minimum.

Fig. 14 shows the ADAAG figure that prescribes wheelchair clearances for doors. Note that the clearance geometries are dependent on the approach of the wheelchair and additional parameters specific to the building element. For example, for doors, the clearance geometry may be dependent on the direction of the swing. For a single swinging door, the ADAAG defines the side from which the user pulls the door to open it as the *pull side* and the side from which the user pushes the door to open it as the *push side*. From each side, the user can approach the opening from the *front*, *hinge side*, or *latch side* of the door:

- For the front pull side approach, the clearance box extends 60 in. (1525 mm) from the wall that contains the opening and the door and covers the width of the opening plus 18 in. (460 mm) on the latch side of the door (left picture of Fig. 14(a)).
- For the front push side approach, the clearance box extends 48 in. (1220 mm) from the wall and covers the

[6] Note that these figures are not shown in this paper.

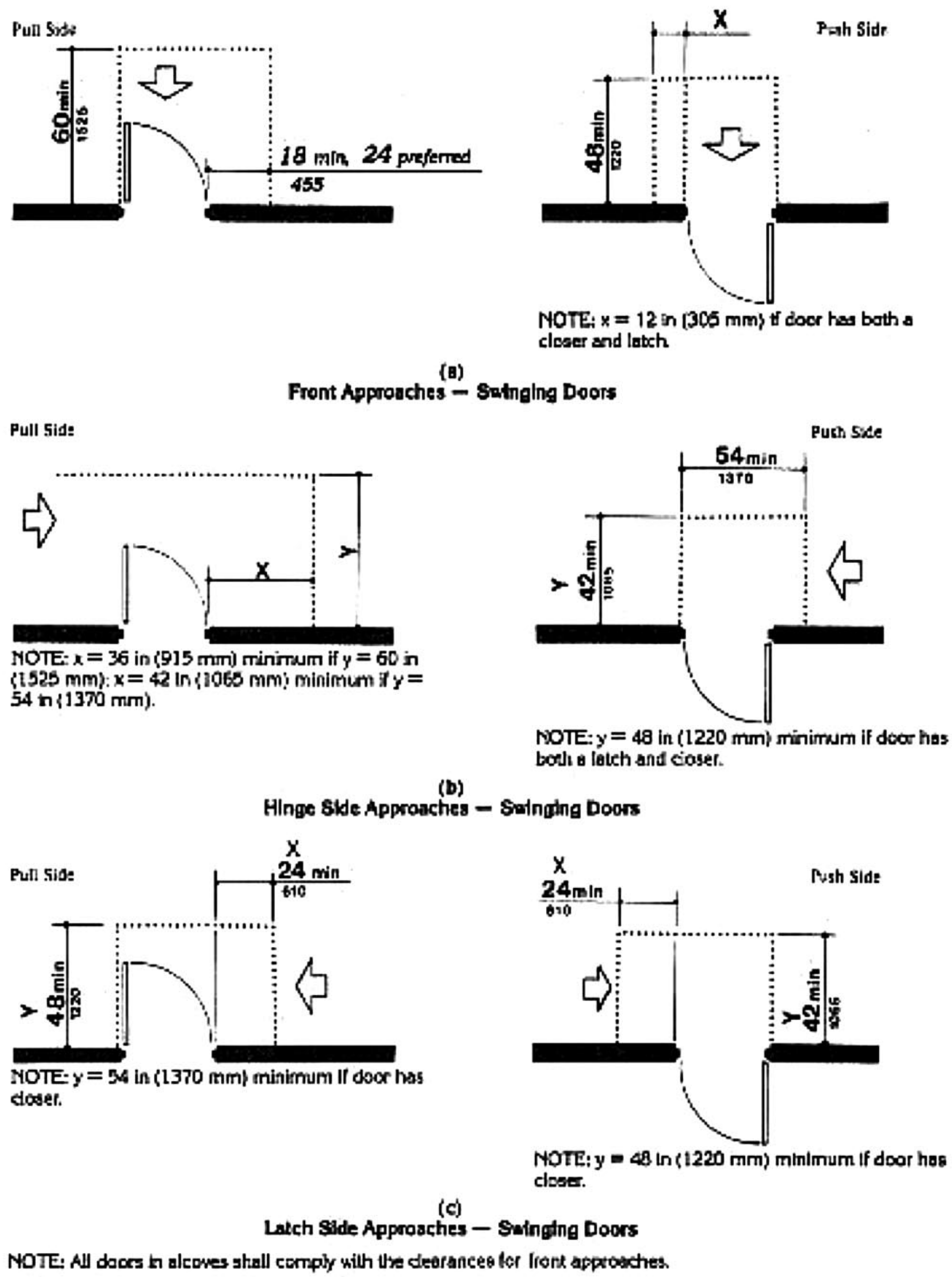

Fig. 14. Door approaches and wheelchair clearance. The motion planner models the different door approaches and clearances, based on the definitions of the ADAAG.

width of the opening plus 12 in. (305 mm) on the latch side if the door has a closer and a latch (right picture of Fig. 14(a)).

- For the hinge pull side approach, the clearance box extends 60 in. (1525 mm) from the wall and covers the width of the opening plus 36 in. (915 mm) on the latch side. Or the clearance box extends at least 54 in. (1370 mm) from the wall and covers the width of the opening plus 42 in. (1065 mm) on the latch side (left picture of Fig. 14(b)).
- For the hinge push side approach, the clearance box extends 42 in. (1065 mm) from the wall (48 in. (1220 mm) if the door has a latch and closer) and covers the width of 54 in. (1370 mm) from the latch side extending toward the hinge side (right picture of Fig. 14(b)).
- For the latch pull side approach, the clearance box extends 48 in. (1220 mm) from the wall (54 in. (1370 mm) if the door has a latch and closer) and covers the width of the opening plus 24 in. (610 mm) on the latch side (left picture of Fig. 14(c)).
- For the latch push side approach, the clearance box extends 42 in. (1065 mm) from the wall (48 in. (1220 mm) if the door has a latch and closer) and covers

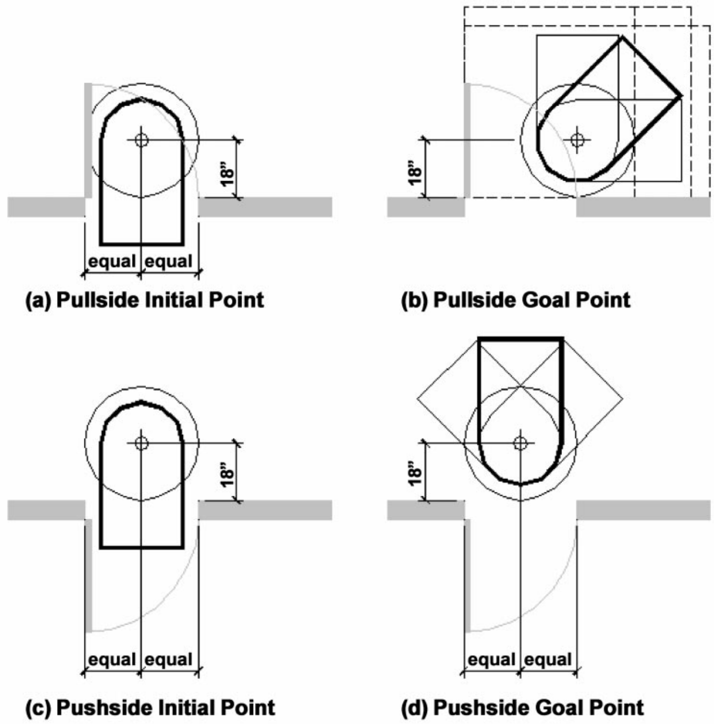

Fig. 15. Initial and goal points for the R_{open} node. The wheelchair moves forward, i.e. up in figures (a) and (c) and down in figures (b) and (d). The goal of one segment, e.g. (b) becomes the initial point of a connected segment, e.g. (c).

the width of the opening plus 24 in. (610 mm) on the latch side (right picture of Fig. 14(c)).

The accessible route analysis examines all possible approaches by performing geometry interference tests on the associated clearance boxes. Failure of all interference tests for either the pull or push side disqualifies the potential R_{open} component. Conversely, if at least one clearance box on either side passes the interference test, the potential R_{open} component qualifies as accessible and assigned as a node in the accessible route graph.

6. Determining the initial and goal points R_{init} and R_{goal} of an accessible segment

Each accessible route segment starts from an initial point and ends at a goal point. To automatically generate and check a route segment using the motion planner, it is necessary to determine the initial and goal positions of the building elements. The following describes the determination of the initial and goal points for certain building elements, such as doors and openings and the toilet.

6.1. Doors and openings

Since the potential opening component is a node in the accessible route graph, the component provides the initial/goal point for a route segment R_{seg}. Fig. 15 illustrates the positions of the initial and goal points associated with the opening. Since the motion planner uses the initial and goal points to generate the potential field in the C_{36} configuration space, the figure shows the circular A_{36} robot as well as the A_{wc} robot. The A_{wc} robot shown in the figure has a fixed orientation associated with the initial points. However, the motion planner accepts any orientation within a 90° range for the orientation of the A_{wc} robot at the goal position. Note that when passing through a door opening, the wheelchair goes from the goal point of a path segment on one side of the door opening to the initial point of another path segment on the opposite side of the door opening. The goal point–initial point sequence through a door opening is either (b)–(c) or (d)–(a) from the figures shown in Fig. 15. The door opening goal point and initial point parameters as shown in Fig. 15 guarantee that a path exists from the goal point–initial point pair.

6.2. Water closet

In general, an R_{goal} node maps to one goal point. However, for certain accessible building elements such as toilet, the motion planner needs to establish more than one goal point to check whether a component is accessible. Fig. 16 illustrates water closet usage by a wheelchair user, an action known as wheelchair transfer. As shown in the figure, the wheelchair user can transfer from the wheelchair to the toilet via two fundamentally different methods: diagonal transfer and side transfer. Thus, the motion planner specifies

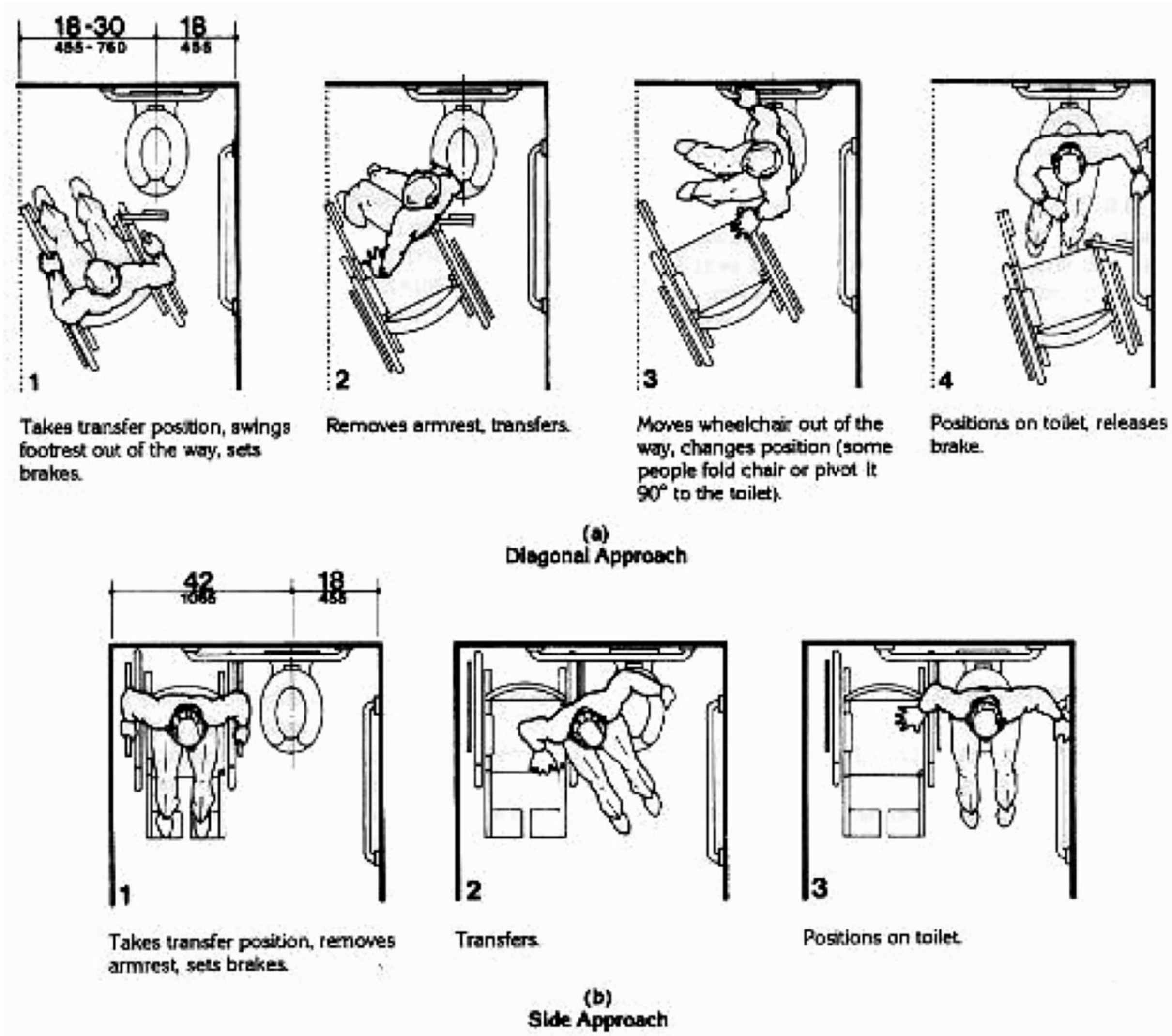

Fig. 16. Wheelchair transfer diagrams for water closets from ADAAG. The motion planner defines both side and diagonal transfer behaviors.

two different goal points and orientations to reflect the different methods.

Fig. 17 illustrates the two goal points and orientations associated with the two transfer options. Note that for the side transfer, the goal point and orientation of the wheelchair robot illustrated in Fig. 17(b) does not directly correspond to the side transfer position illustrated in Fig. 16(b). Currently, the motion planner restricts the wheelchair to only forward motion, and the ADAAG assumes backing up to the final side transfer position. Therefore, the motion

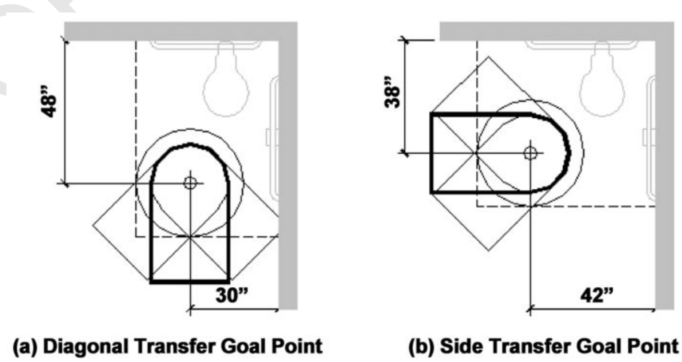

Fig. 17. Goal points for diagonal and side transfers.

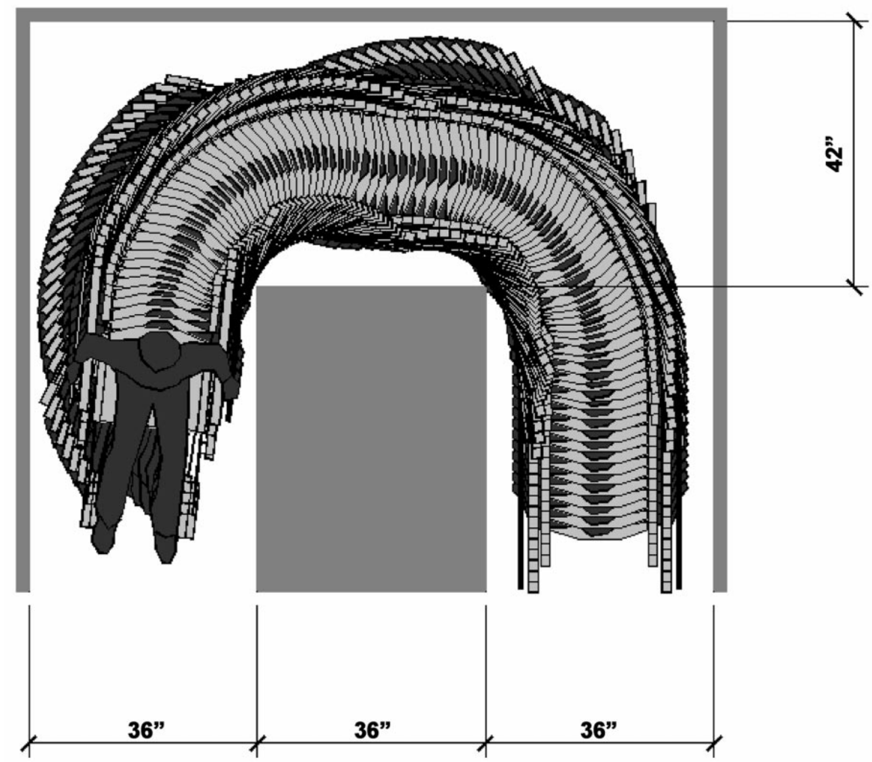

Fig. 18. Example of an U-turn-around an obstacle exception. This trace of a wheelchair path around an obstruction illustrates that the performance-based path-planner can determine that a path can be accessible, although non-compliant with the prescriptive code.

planner positions the wheelchair robot such that it is possible to make the backup move to the final side transfer position.

7. Prescriptive code analysis and performance-based usability analysis

Because of the prescriptive nature of the disabled access code, it cannot address all possible building design configurations or wheelchair use patterns; the code limits the special cases it addresses to the turn-around-an-obstruction exceptions. In practice, wheelchair users can comfortably use a large number of design configurations that do not comply with the prescriptive accessible route provisions from the ADAAG. On the other hand, a design configuration which is code complied does not necessarily imply accessible. Here, we analyze design configurations against the prescriptive parameters as given in the ADAAG and compare the results to the performance-based analysis based on the motion planner.

For a given configuration, the following four scenarios for accessibility are possible:

- Code-compliant and usable.
- Not code-compliant and usable (Section 7.1).
- Code-compliant and unusable (Section 7.2).
- Not code-compliant and unusable.

The performance-based analysis uses specific provisions from the ADAAG to instantiate the turning radius parameters, and by default, the tested configurations were both code-compliant and usable. Providing examples that are both non-compliant and unusable can also be trivially demonstrated, for example, with a less-than 36 in. wide (915 mm wide) corridor. The following examples illustrate a non-compliant route that a wheelchair user can actually negotiate and a code-compliant route that a wheelchair user cannot negotiate.

7.1. Example 1

The first example presents a design configuration illustrated in Fig. 18 that falls under the U-turn-around-an-obstacle exception category: the width of the obstruction is less than 48 in. (1220 mm), and the configuration cannot be transformed into the 90°-turn-around-an-obstacle exception by making the obstruction wider than 48 in. (1220 mm). Following the parameters of the ADAAG Provision 4.3.3, the configuration fails to comply with the exception that:

- The widths of the first and third legs are less than 42 in. (1065 mm).
- The width of the second leg is less than 48 in. (1220 mm).

Using the performance-based parameters for the turning

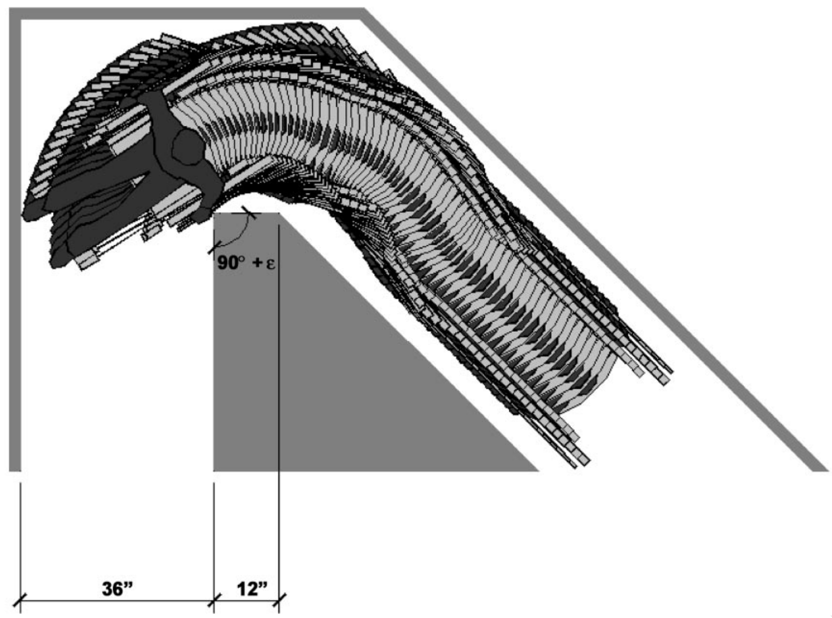

Fig. 19. Example of a code-compliant but unusable configuration. This design example is accessibility-compliant, but the motion planner shows that it is unusable.

radii established in the performance-based analysis, the motion planning simulation returns a successful path τ around the obstruction for the non-compliant configuration as illustrated in the figure. Thus, from the performance perspective, the motion planner deems that the configuration is usable by a wheelchair user.

This example demonstrates that the wheelchair model constructed from the constraints of the two ADAAG U-turn configurations can successfully navigate through a non-compliant U-turn configuration—however, such a demonstration simply points out that the code might be too conservative.

7.2. Example 2

Fig. 19 shows a configuration that is a code-complied but unusable situation. Following the parameters given in the ADAAG Provision 4.3.3, the design complies with the code in that:

- The accessible route is equal to or greater than 36 in. wide.
- Since it is not a 180° U-turn, the turn-around-an-obstruction exception does not apply.

Using the performance-based parameters, the motion planning simulation fails to return a path τ around the obstruction for the code-compliant configuration. By trial and error, one can extend the length of the second leg to find a usable design configuration shown in Fig. 20.

This example illustrates ambiguities that exist in current prescriptive code. First, note that if the angle between the second and third leg equals 90° instead of exceeding 90°, the first turn-around-an-obstruction exception from Provision 4.3.3 might apply. The building official may contend that the exception applies with a small angular increment ϵ, but as ϵ grows, the configuration does not qualify for the exception. The ambiguity of at what point the prescribed configuration applies illustrates the difficulty of applying a prescriptive-based code directly. On the other hand, by changing the angle between the second and third legs of the route from $90° + \epsilon$ to 90° or $90° - \epsilon$, the motion planner demonstrates the overly restrictive nature of the 90°-turn-around-an-obstruction exception from Provision 4.3.3. While not explicitly stated, the exception should apply to angles less than 90° since this configuration would constitute a more difficult accessible route. Furthermore, as illustrated in Fig. 20, the second and third leg dimensions provide a viable path τ around the obstruction, and these lengths are clearly less than the required 48 in./42 in. (1220 mm/1065 mm) exception requirement, as in the provision.

8. Accessibility analysis of floor layout

We applied the methodology to analyze the accessibility of the floor plan for an existing building as shown in Fig. 21. Fig. 22 shows the analysis report with a view of the modeled floor plan [3]. The comments associated with inaccessible building components have links to the prescriptive provisions of the ADAAG document as an informative guide.

The analysis reports shown in Fig. 22 reports that there is no accessible route to the water closet in the men’s bathroom, and thus there is no accessible toilet in the building.

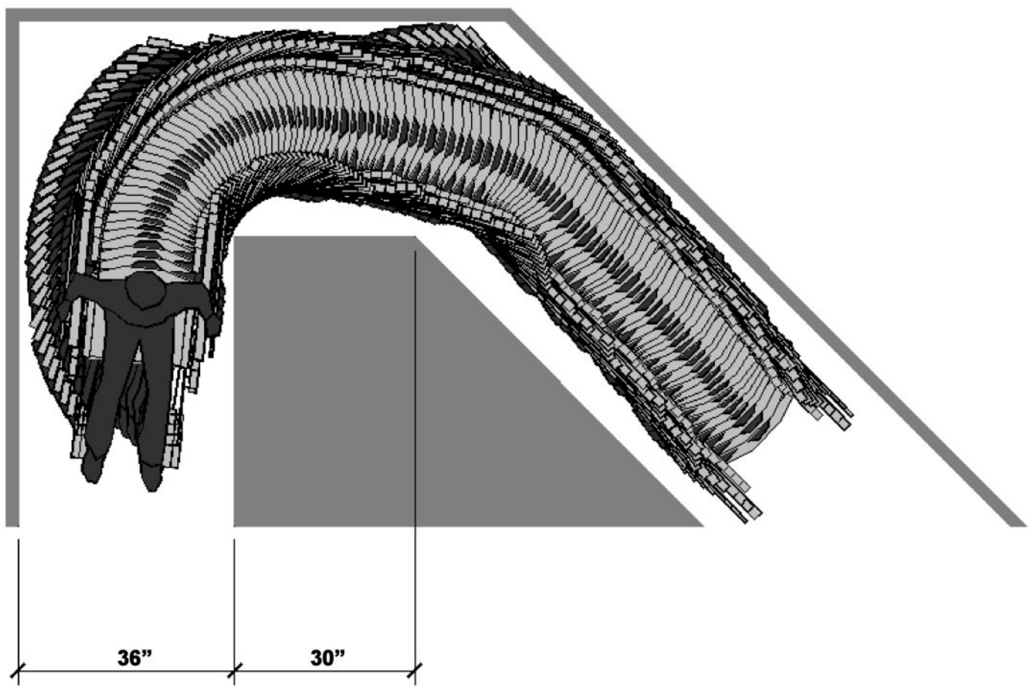

Fig. 20. Example of a viable but non-compliant path around an obstruction. With the motion planner incorporated into a design system, a designer could recognize a problem such as that of Fig. 19, change the building design model, and immediately use the path planner to assess usability of design variants, such as this one.

Fig. 23 confirms the inaccessibility of the water closet. Here, the wheelchair user is not able to pass through the stall's doorway. It is interesting to note that the partition walls were added to the original plan to ensure privacy for the toilet user. Ironically, the addition of these walls made the toilet inaccessible to wheelchair users. With the removal of the partitions, the men's bathroom would revert back to a single-occupancy from a multiple-occupancy. As shown in Fig. 24, without the partition walls, the motion planner generates an accessible route to the toilet. Note that the

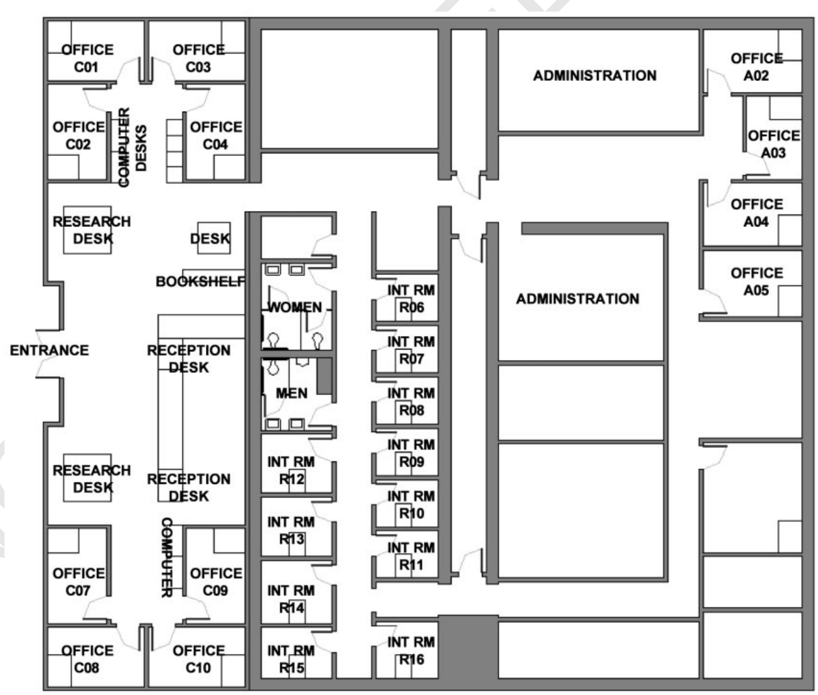

Fig. 21. Floor plan of an actual facility. As shown in Fig. 22, the motion planner found that it lacked wheelchair accessibility to the men's toilet. As shown in Fig. 23, the men's toilet is indeed inaccessible to a wheelchair user.

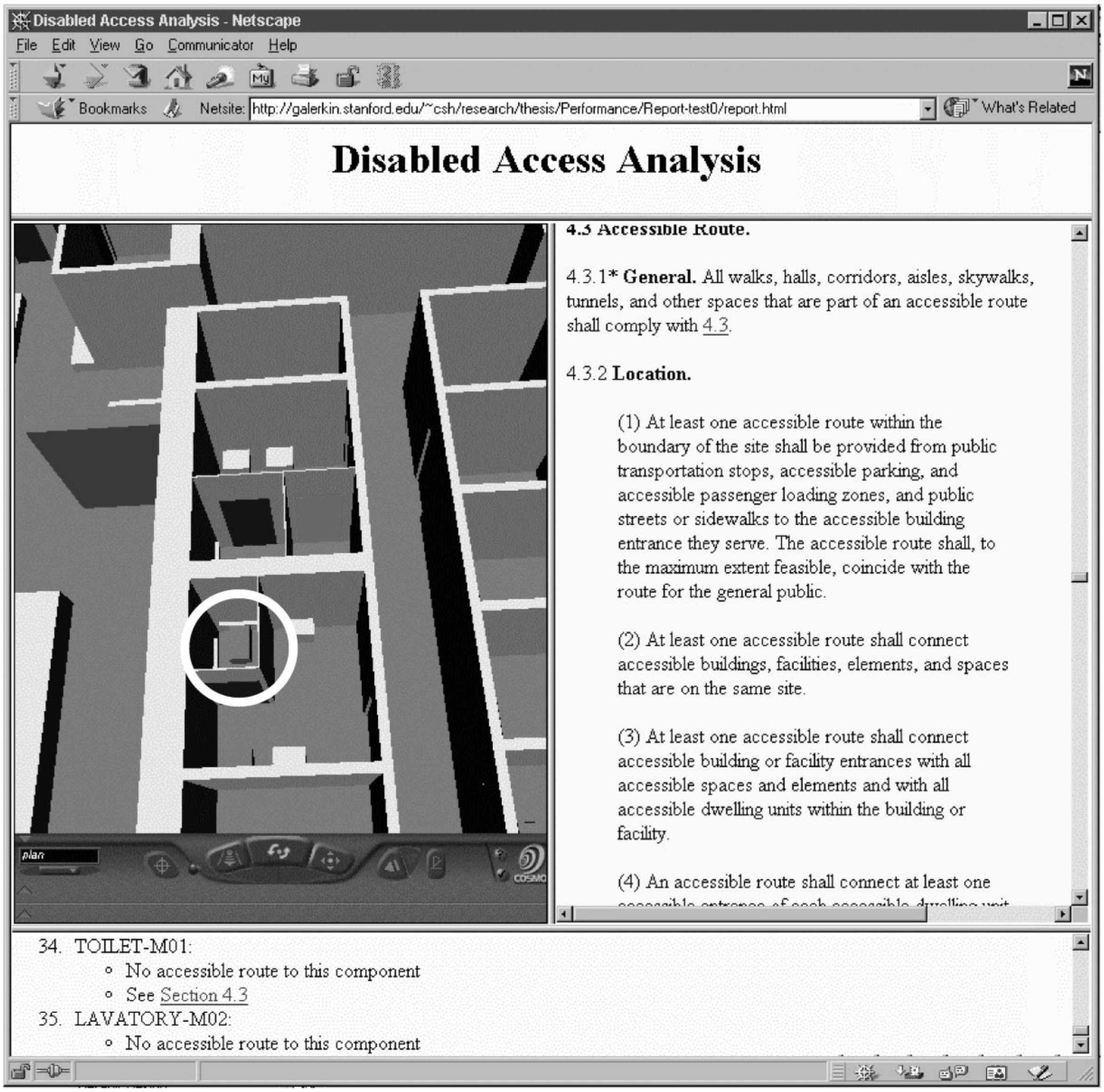

Fig. 22. Accessibility analysis report. This automatically generated report circles the inaccessible men's toilet (left pane) and identifies the relevant section of the ADAAG code (right pane).

path is discontinuous between adjacent spaces since the route segments are generated between building elements and the openings and entrances are checked independently. We performed similar analyses to check access to other facilities such as the bookshelf, the women's bathroom and the interview room, etc. [4].

9. Summary and discussion

This paper discusses the nature and benefits of performance-based analysis of wheelchair accessibility of a facility using motion planning techniques developed in robotics research. In the practices of architects, wheelchair designers and wheelchair users and in its computer implementation, wheelchair accessibility is a knowledge-intensive activity. This paper discusses one approach to implementation of knowledge-intensive engineering analysis in the computer and our results in applying this approach. Built on the practice of architecture, the design of building codes, theory of robotic motion planning and building product models, this work is an example of the multidisciplinary engineering informatics methods that have started to demonstrate high performance in applications of computers in engineering. Traditional expert systems have tried to replicate the knowledge-intensive practices of practitioners, but their implementations have often proved to be ad hoc in design and brittle in performance [9].

The performance-based motion-planning techniques developed directly capture motion and behavior given the

Fig. 23. Inaccessible water closet. The photo shows that a wheelchair user is unable to access the men's toilet in the facility diagrammed in Fig. 21 and referenced in Fig. 22.

wheelchair's parameters as described in the ADAAG. This direct performance analysis obviates the need for the complicated exception analysis associated with the ADAAG accessible route parameters, an artifact that is a consequence of the prescriptive nature of the ADAAG. Furthermore, the performance-based analysis method can ensure the usability of an accessible route. The ADAAG prescribes minimal legal requirements. This general prescription necessarily ignores details of individual wheelchair designs and the abilities and preferences of individual users. Thus, the prescriptive ADAAG can inform the design of wheelchairs by manufacturers, and it can inform the design of individual buildings by clients, but it cannot represent their specific situation. The detailed behavior model

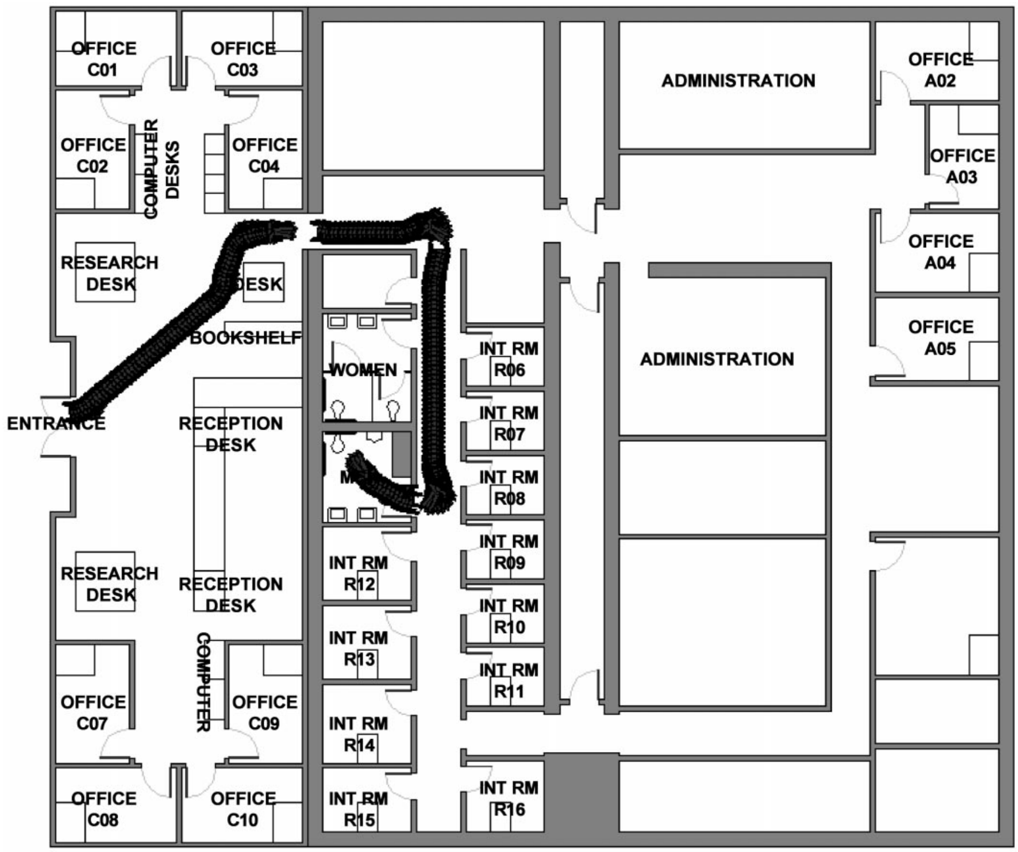

Fig. 24. Wheelchair route to men's toilet with the stall partitions removed. Integrated into a design system, the designer can make what-if changes to the design and invoke the motion-planner to analyze wheelchair accessibility of each design option.

and simulation can readily accommodate the behavior details of different wheelchair designs, and designers can also use this method to analyze the performance of different wheelchairs in different building designs while considering the detailed abilities and preferences of different users.

The development method created a number of specific instances of very general system design components, including:

- Explicit symbolic (non-numeric) representation of the physical components of the (building) design and some component attributes and relationships. The model explicitly represents fundamental concepts of the engineering problem: geometric forms (of both the building and wheelchairs), the design functional intent (of building components, i.e. that certain kinds of building components are to be wheelchair accessible, and wheelchairs, i.e. the assumption that motion is always forward and that turning radii are constrained), and computable behaviors (of a wheelchair and a building, i.e. paths and wheelchair accessibility). The simulator model uses this model of the user and its functional behaviors.
- Both to aid development and understanding of the analysis, the simulation system has an associated graphical user interface that shows 3D views of the designed system and time-varying animations of the behavior of the physical system user, as well as helpful explanations of the findings of the analysis system.
- Interactive in system use, enabling system users (both designers and, potentially, code checkers) to interpret the behavior of any design version and change the product and process models, exploring predicted performance of designs using criteria that are difficult to model explicitly.

The performance-based accessible route analysis uses the dimension for a wheelchair as given by the ADAAG to develop the A_{wc} robot parameters. The prescriptive nature of the code creates an indirect relationship between provision parameters and the wheelchair dimensions and behavior. In fact, the relationship between the actual wheelchair constraints and the prescriptive route usability analysis is not explicitly defined. In contrast, the approach described can model more accurately the desired performance and usability of space.

A designer can vary the parameters used by the motion planner that is developed in this paper. Varying specific parameters allows wheelchair manufacturers and users to test the behavior of a specific wheelchair model or assign personal preferences. Varying the A_{36} robot's diameter (and influencing the C_{36} configuration space) allows the user to choose a preferred comfort level for the path width independent of the actual wheelchair parameters. Of course, the diameter should exceed the wheelchair width. Manufacturers make wheelchairs with various physical dimensions, and the performance-based analysis can easily capture these dimensions to model the A_{wc} robot. In addition, the turning radii and the centerpoint of the turn depend on the wheelchair's mechanical constraints. Finally, independent of these mechanical constraints, users have their own comfort level associated with the possible turning radii r_1 and r_2.

In short, the performance-based approach allows wheelchair manufacturers and users to specify custom wheelchair parameters. Once these parameters are determined, the motion planner can simulate wheelchair movement through a space that has been deemed usable by a 'model' or 'general-purpose' wheelchair to confirm the custom wheelchair's usability.

This paper has illustrated the use of the performance prediction method using both small illustrative examples and a real commercial example. Code specifying bodies have started to use performance-based methods for some aspects of building performance, e.g. energy. For example, the US Department of Energy Building Standards and Guidelines Program (BSGP) is a software tool that checks compliance with the commercial building energy code using whole-building performance simulation methods.

In this work, the non-holonomic path-planning analysis developed limits the wheelchair motion to three options: left, right, and forward. The motion planner imposes this restriction in an attempt to capture the intent of the prescribed code and match the compliance results of the ADAAG accessible route provisions. However, the technique can be extended to accommodate a number of options of forward motion to include various (left or right) and forward turns. In general, we implemented a simple motion planner: it demonstrates that the performance-based simulation approach is interesting and may lead to a change how designers and code checkers specify and check access requirements.

The motion planner described in this paper focuses on forward motion alone. However, the motion planner can be extended to support backward motion. Reeds [10] describes optimal paths for a car-like robot that can reverse its direction. Indeed several ADAAG provisions assume backward motion. The motion planner would limit the number of backups for a given path τ according to the specifications of the building code or the user.

While this study has illustrated the benefits for a performance-based analysis of accessible route, further possible extensions of current works warrant. First, while the motion planner guarantees the discovery of the 36 in. (915 mm) wide path if one exists, the non-holonomic planner developed for the actual wheelchair motion for the second pass might not find a path even if one exists. The problem concerns the discretization of the configuration space. Determining the necessary discretization granularity for the non-holonomic configuration space is not straightforward since the location of the wheelchair robot's next possible position (using trigonometric functions) may not correspond to the exact grid discretization. A promising alternative is the randomized motion planner for robots developed under kinematic and dynamic constraints in which the probability that the planner fails to find a path when one exists converges toward zero [5].

10. Conclusions

This work is an example of the kind of automated analysis of the predicted performance of a design that modern automated simulation now enables. The particular performance analysis is complicated, highly knowledge-intensive, and difficult both in theory and in practice. There are very general components to the development method described in this paper and summarized in the discussion. These components build on carefully identified and selected and represented engineering knowledge. The computer implementation of this model-based simulation provides a new way to predict the performance of designed systems, one that appears to be significantly more powerful than either manual or static heuristic analyses of predicted system behavior. The system components seem general in the sense that they potentially enable many kinds of high-performance predictions of system behavior. Although the analysis system of this paper stores the design model in an application-specific database, a more general implementation could store both the design and the user (wheelchair) models in shared, reusable and potentially changing databases.

10. Uncited references

[1]. [2].

Acknowledgements

This research is supported by the National Science Foundation under Grant No. EIA-9983368 and by the Center for Integrated Facility Engineering at Stanford University. Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the sponsoring agencies. The authors would like to thank Mr Joe Cavanaugh for his participation in this research and the thoughtful anonymous reviewers.

References

[1] ADAAG. Americans with disabilities act accessibility guide. Washington, DC: Access Board, US Architectural and Transportation Barriers Compliance Board, 1997.

[2] CSC. Energy efficiency standards for residential and nonresidential buildings. Sacramento, CA: California Energy Commission, 1999 title 24, section 6.

[3] Han CS, Kunz JC, Law KH. A client/server framework for on-line building code checking. J Comput Civil Engng, ASCE 1998;12(4): 181–94.

[4] Han CS. Computer models and methods for a disabled access analysis design environment. PhD Thesis, Department of Civil and Environmental Engineering, Stanford University, Stanford, CA, 2000.

[5] Hsu D, Kindel R, Latombe JC, Rock S. Randomized kinodynamic motion planning with moving obstacles. Workshop on Algorithmic Foundations of Robotics, 2000.

[6] Kiliccote H. A standards processing framework. PhD Thesis, Department of Civil and Environmental Engineering, Carnegie Mellon University, Pittsburgh, PA, 1996.

[7] Latombe JC. Robot motion planning. Norwell, Mass: Kluwer Academic Publishers, 1991.

[8] Latombe JC. A fast planner for a car-like indoor mobile robot. Proceedings of the Ninth National Conference on Artificial Intelligence, Anaheim, CA, 1991b. p. 659–65.

[9] Partridge D. The scope and limitations of future generation expert systems. Future Generations Computer Systems 1987;3:1–10.

[10] Reeds JA, Shepp RA. Optimal paths for a car that goes both forward and backward. Pacific J Math 1991;145(2):367–93.