Rapid global path planning algorithm for unmanned surface vehicles in large-scale and multi-island marine environments

Dong Wang [1, 2] , Jie Zhang [Corresp., 1, 2] , Jiucai Jin [Corresp., 2] , Deqing Liu [2] , Xingpeng Mao [1]

[1] School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin, Heilongjiang, China

[2] First Institute of Oceanography, Ministry of Natural Resources, Qingdao, Shandong, China

Corresponding Authors: Jie Zhang, Jiucai Jin
Email address: zhangjie@fio.org.cn, jinjiucai@fio.org.cn

A global path planning algorithm for unmanned surface vehicles (USVs) with short time requirements in large-scale and complex multi-island marine environments is proposed. The fast marching method-based path planning for USVs is performed on grid maps, resulting in a decrease in computer efficiency for larger maps. This can be mitigated by improving the algorithm process. In the proposed algorithm, path planning is performed twice in maps with different spatial resolution (SR) grids. The first path planning is performed in a low SR grid map to determine effective regions, and the second is executed in a high SR grid map to rapidly acquire the final high precision global path. In each path planning process, a modified inshore-distance-constraint fast marching square (IDC-FM^2) method is applied. Based on this method, the path portions around an obstacle can be constrained within a region determined by two inshore-distance parameters. The path planning results show that the proposed algorithm can generate smooth and safe global paths wherein the portions that bypass obstacles can be flexibly modified. Compared with the path planning based on the IDC-FM^2 method applied to a single grid map, this algorithm can significantly improve the calculation efficiency while maintaining the precision of the planned path.

1   **Rapid global path planning algorithm for unmanned surface**

2   **vehicles in large-scale and multi-island marine environments**

3   Dong Wang[1,2], Jie Zhang[1,2], Jiucai Jin[2], Deqing Liu[2], Xingpeng Mao[1]

4   [1] School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin,
5   Heilongjiang, China

6   [2] First Institute of Oceanography, Ministry of Natural Resources, Qingdao, Shandong, China

8   Corresponding Author:

9   Jie Zhang[1,2], Jiucai Jin[2]

10 No. 6 Xianxialing Road, Laoshan District, Qingdao, Shandong, 266061, China

11 Email address: zhangjie@fio.org.cn, jinjiucai@fio.org.cn

14 **Abstract**

15 A global path planning algorithm for unmanned surface vehicles (USVs) with short time
16 requirements in large-scale and complex multi-island marine environments is proposed. The fast
17 marching method-based path planning for USVs is performed on grid maps, resulting in a
18 decrease in computer efficiency for larger maps. This can be mitigated by improving the
19 algorithm process. In the proposed algorithm, path planning is performed twice in maps with
20 different spatial resolution (SR) grids. The first path planning is performed in a low SR grid map
21 to determine effective regions, and the second is executed in a high SR grid map to rapidly
22 acquire the final high precision global path. In each path planning process, a modified inshore-
23 distance-constraint fast marching square (IDC-FM^2) method is applied. Based on this method,
24 the path portions around an obstacle can be constrained within a region determined by two
25 inshore-distance parameters. The path planning results show that the proposed algorithm can
26 generate smooth and safe global paths wherein the portions that bypass obstacles can be flexibly
27 modified. Compared with the path planning based on the IDC-FM^2 method applied to a single
28 grid map, this algorithm can significantly improve the calculation efficiency while maintaining
29 the precision of the planned path.

30 **Introduction**

31 Research on unmanned surface vehicles (USVs) has received increased attention in various
32 military and civilian applications over recent years (*Yan et al., 2010; Campbell, Naeem & Irwin,*
33 *2012; Liu et al., 2016*). Robust and reliable navigation, guidance, and control (NGC) systems are
34 required for USVs to perform a variety of complex marine missions. Path planning is an essential

35   component of an NGC system. The main aim of path planning is to calculate a collision-free path

36   with specific requirements. It determines the automation level of USVs and ensures the

37   reliability and success of missions (*Tan et al., 2020*).

38   Global path planning is an important aspect of path planning. A variety of algorithms for the

39   global path planning of USVs have been studied for different requirements. Dijkstra's algorithm

40   (*Dijkstra, 1959*) is a classic graph-based algorithm that can plan the shortest global path for

41   USVs (*Xie et al., 2016; Singh et al., 2018*). As an improvement of Dijkstra's algorithm, the A*

42   algorithm (*Hart, Nilsson & Raphael, 1968*) and its related improved algorithms are also

43   commonly used in the global path planning of USVs (*Campbell, Naeem & Irwin, 2012*), such as

44   the direction priority sequential selection method (*Naeem, Irwin & Yang, 2012*); the A*

45   algorithm, which considers environment effects (EEA*) (*Lee et al., 2105*); and the ARC-Theta*

46   algorithm, which considers both the angular rate and heading angle of USVs (*Kim et al., 2014*).

47   These algorithms exhibit acceptable convergence and consistency; however, the planned paths

48   must be further smoothed. The artificial potential field (APF) method (*Khatib, 1986*) is a classic

49   method for both global and local path planning. The main drawback of the traditional APF

50   method is the local minimum problem. Therefore, different studies (*Guo, Gao & Cui, 2013;*

51   *Song, Hao & Su, 2020*) have been conducted to solve the local minimum problem for the path

52   planning of robots or USVs. Evolutionary methods such as the genetic algorithm (GA) (*Kim et*

53   *al., 2017; Arzamendia et al., 2019; Wang et al., 2020*), particle swarm optimization (PSO) (*Song*

54   *et al., 2015*), and ant colony optimization (ACO) (*Song, 2014; Xia et al., 2019*) have also been

55   applied in the path planning of USVs. GA is robust and adaptable, but it has the shortcomings of

56   poor local search ability and the premature convergence phenomenon. Therefore, other methods,

57   such as the simulated annealing algorithm, are usually introduced to improve the performance of

58   GA (*Zhang, Xu & Xie, 2019*). Nevertheless, the path planned by GA still lacks consistency.

59   Similar to GA, PSO has the advantage of strong robustness and the shortcoming of premature

60   convergence. However, it is dependent on various parameters, and there is no specific theory for

61   guiding the setting of these parameters for different problems. Similarly, the parameter selection

62   of ACO is more dependent on experience, and this algorithm can easily fall into local extremum.

63   In some special applications, the shortest time may be an important requirement for USVs.

64   The fast marching method (FMM) can be a solution for time-optimal global path planning. This

65   method was first proposed by *Tsitsiklis (1995)*, and *Adalsteinsson & Sethian (1995)*

66   independently and was extended by *Sethian (1999)*. The path planned by the FMM is usually

67   extremely close to the obstacles. One solution is to adjust the speed map, as exemplified by the

68   method with an adjusted cost function (*Messias et al., 2014*) and the FM^2 method (*Garrido et al.,*

69   *2007*). FMM-based methods have been widely used in path planning applications (*Gómez et al.,*

70   *2013; Amorim & Ventura, 2014; Alvarez et al., 2015; González et al., 2016*). Marine

71   applications based on FMM were introduced by *Garrido, Alvarez & Moreno (2020)*. Interesting

72   modifications for the FM^2 method have been performed, and the FMM has been subjected to a

73   vector field considering the effects of several vector variables such as wind flow or water

74   currents (*Garrido, Alvarez & Moreno, 2020*). In addition, studies on path following and obstacle

75   avoidance and formations have also been conducted using FMM (*Garrido, Alvarez & Moreno,*
76   *2020*). USV formation path planning has also been performed by *Liu & Bucknall (2015)* and *Tan*
77   *et al. (2020)*, and an angle-guidance FM^2 method has been used for the Springer USV to make
78   the generated path compliant with the dynamics and orientation restrictions of USVs (*Liu &*
79   *Bucknall, 2016; Liu, Bucknall & Zhang, 2017*). An improved anisotropic fast marching method
80   using a multi-layered fast marching was proposed by *Song, Liu & Bucknall (2017)*, which
81   combines different environmental factors and provides interesting results. In addition to the
82   global path planning, the FMM-based methods also show potential in collision avoidance of
83   USVs (*Wang, Jin & Er, 2019; Garrido, Alvarez & Moreno, 2020*). These successful studies have
84   demonstrated the potential of FMM-based methods in global path planning of USVs.
85   The basic FMM can plan time-optimal paths for USVs. However, some of the main
86   shortcomings is that the paths planned by the FMM are too close to obstacles, and there may be
87   abrupt turns when the paths bypass obstacles with sharp corners. Thus, the FM^2 method is
88   proposed to address these problems, and two dimensionless parameters are introduced to adjust
89   the paths more flexibly (*Garrido et, al. ,2007*; *Garrido, Alvarez & Moreno, 2020*). However, the
90   two introduced parameters lack clear physical meaning, and the suitable adjustment is difficult to
91   identify with specific values. Moreover, the adjustment effects of these two parameters are not
92   common because the adjustment degree with the same parameters varies with different grid
93   maps. Another common problem about FM-based methods is that the computational efficiency
94   of path planning decreases sharply when the scale of the grid map is very large. Therefore, we
95   make two improvements to address the mentioned shortcomings of the basic FM^2 method. First,
96   we introduced an inshore-distance-constraint fast marching square (IDC-FM^2) method to
97   improve the inshore path adjustment performance for the first time. Comparing with the basic
98   FM^2 method, the IDC-FM^2 method applies two inshore distance parameters other than the two
99   dimensionless parameters to adjust the paths around the obstacles. The adjustment effects for
100 path planning of the IDC-FM^2 method are stable in different grid maps as the IDC-FM^2 method
101 can constrain the path portions around the obstacles within the region constrained by the two
102 inshore distance parameters. Further, to improve the computational efficiency, the algorithm that
103 applies the IDC-FM^2 method based on two-level spatial resolution grid maps is designed.

Related Methods

Environment map model

106 In 2D global path planning applications based on the FMM or its improved methods, discrete
107 numerical calculations are based on cartesian grid maps. Therefore, an environment map should
108 first be converted into a binary grid map with a suitable spatial resolution (SR). Free and open-
109 source satellite images (such as Google satellite images) can be used as the data source for the
110 maps in most USV applications, and the corresponding grid maps can be generated using image
111 processing (*Shi et al., 2018*) combined with manual assistance. The binary grid is set as an
112 obstacle grid when an obstacle exists at the geographic location (0 values); otherwise, it is set as
113 a free grid (1 value).

114 **Fast marching square method**

115 When using FM-based methods to plan a path, the basis is the FMM. The core work of the FMM

116 is calculating solutions of the Eikonal equation (*Tsitsiklis, 1995*; *Adalsteinsson & Sethian, 1995*).

117 The Eikonal equation describes a wave front propagation scenario from sources with the speed of

118 a wave front given as F_x at cell x. It can be expressed as $\|\nabla T_x\| F_x = 1$, where T_x is the arrival

119 time of the wave from the source to the cell x and ∇ is a vector differential operator. From the

120 perspective of time-cost, the Eikonal equation can be expressed in another form (that of *Lin,*

121 *2003*):

122 $$\|\nabla T_x\| = \tau_x \tag{1}$$

123 where τ_x is the time-cost at cell x and is equivalent to $1/F_x$. The solution we want to calculate is

124 T_x, and all of them compose the arrival time map, $\mathcal{T}(x)$. All time-cost τ_x compose the time-cost

125 function map, $\tau(x)$.

126   The FMM was first proposed independently by *Tsitsiklis (1995)* and *Adalsteinsson & Sethian*

127 *(1995).* The solution T_x in cell x can be interpreted as the wave arrival time from the nearest

128 source to cell x. Three cell sets (the accepted set $\mathcal{S}_A$, trial set $\mathcal{S}_T$, and far set $\mathcal{S}_F$) are defined to

129 calculate $\mathcal{T}(x)$. The set of all cells at which T_x will not change is $\mathcal{S}_A$. $\mathcal{S}_T$ is the set of cells to be

130 examined. Every T_x in $\mathcal{S}_T$ has been previously computed but may be updated. $\mathcal{S}_F$ is the set of all

131 other cells with T_x, which has never been computed (*Lin, 2003*).

132   The calculation process includes initialization and loop procedures. During initialization,

133 source cells x_0 with $T_{x_0} = 0$ are set to $\mathcal{S}_T$, and all other cells are set to $\mathcal{S}_F$ with $T_x = \infty$. After the

134 initialization, the cell x_m with the smallest T value in $\mathcal{S}_T$ is selected and moved from $\mathcal{S}_T$ to $\mathcal{S}_A$.

135 Thereafter, the non-accepted neighbor cells of x_m are updated, including the solutions and cell

136 sets. Considering Tsitsiklis's deduction (*Tsitsiklis, 1995; Lin, 2003*), each of the new neighbor

137 solutions is determined by:

138 $$T_x = \min\left(\tilde{T}_x, \left(T_{x_m x_i}, i = 1, 2\right)\right) \tag{2}$$

139 where $\tilde{T}_x$ denotes the original solution. $T_{x_m x_i}, i = 1, 2$ are the candidate solutions for the paths that

140 pass through the line segment $x_m x_i$ and propagate to cell x (see *Fig. 1*), which are calculated by:

141 $$T_{x_m x_i} = \begin{cases} \frac{1}{2}\left(T_{x_m} + T_{x_i} + \sqrt{2\tau_x^2 - \left(T_{x_m} - T_{x_i}\right)^2}\right) & ,T_{x_m x_i} > T_{x_m}, \text{and } T_{x_m x_i} > T_{x_i} \\ \min\left(T_{x_m}, T_{x_i}\right) + \tau_x & ,\text{others} \end{cases} \tag{3}$$

142 where T_{x_m} and T_{x_i} are the accepted solutions at cells x_m and x_i, respectively. If cell x_i is in $\mathcal{S}_T$,

143 T_{x_i} is ∞. For the cell set, if a non-accepted neighbor cell x is in $\mathcal{S}_F$, it moves from $\mathcal{S}_F$ to $\mathcal{S}_T$. The

144   loop procedure is performed continually until all cells are in S_A. The resulting map is the arrival

145   time map, $\mathcal{T}$.

146   The main disadvantage of the basic FMM is that the computed path is too close to obstacles

147   and forces the vehicle to perform abrupt turns (*Garrido, Alvarez & Moreno, 2020*). As described

148   in *Garrido et, al.(2007)* and *Garrido, Alvarez & Moreno, (2020)*, a smooth path with sufficient

149   safety distances from obstacles can be computed using the FM² method. This method applies the

150   basic FMM twice. The procedure for computing paths is described as follows (*Garrido, Alvarez*

151   *& Moreno, 2020*):

152   1.   The environment is modeled as a binary grid map (*Fig. 2A*).

153   2.   The FM-1ˢᵗ step. All obstacle cells are used as wave sources ($T = 0$), expanding several

154   waves at the same time at a constant speed. The value of each cell in the resulting map

155   indicates the time required for a wave to reach the closest obstacle (see *Fig. 2B*). This is

156   proportional to the distance from the obstacles. By reversing the meaning of these values,

157   they can be understood as the maximum admissible speed at each cell. Finally, the speed

158   values are rescaled to fix a maximum cell value of 1.

159   Modifications of the speed map with two parameters, α and β, can adjust distances

160   between the computed paths and obstacles. The value of each cell in the speed map $F_{i,j}$ is

161   adjusted exponentially by α:

162    $$newF_{i,j} = F_{i,j}^{\alpha} \tag{4}$$

163   The parameter β is used to saturate the values in the speed map. It is defined within the

164   range of 0 and 1. Every $F_{i,j}$ with a value greater than β is set to one (see *Fig. 2C*).

165   Comparing with a path without modifications, the path modified by α will be closer to

166   obstacles when $\alpha < 1$. On the contrary, if $\alpha > 1$, the modified path will stay further away

167   from obstacles. The parameter β allows the path to move closer to obstacles. When β is

168   smaller, the modified path will be closer to obstacles.

169   3.   The FM-2ⁿᵈ step. The goal point is used as a unique wave source. The wave is expanded

170   over the map until the starting point is reached. The speed at each cell (equivalent to $1/\tau_x$) is

171   obtained from the modified speed map computed in the FM-1ˢᵗ step. The resulting arrival

172   time map is shown in *Fig. 2D*.

173   4.   Finally, a gradient descent is applied over the resulting arrival time map from the starting

174   point to the goal point. An optimal path in terms of the arrival time, smoothness, and safety

175   is obtained.

176   **Gradient descent method**

177   The global path can be extracted by applying the gradient descent method. The path propagates

178   along the gradient descent direction from the starting position P_s to the goal position P_g with a

179   step length d. The value of d can be set to a value equal to the SR of the grid map. The gradient

180   of the cell $x_{i,j}$ is calculated by:

181 $\quad \nabla T_{i,j} = \left[\frac{T_{i+1,j} - T_{i-1,j}}{2} \quad \frac{T_{i,j+1} - T_{i,j-1}}{2} \right]^{\mathrm{T}}$ (5)

182 where $T_{i,j}$ represents the arrival time value at grid $x_{i,j}$ and the non-italic T in the upper-right

183 corner represents the transpose of vector $[x \ y]$.

184 $\quad$ The path waypoints are not limited to cell centers. Generally, the gradient of a path waypoint

185 $P_n = (u,v)$ located in the cell $x_{i,j}$ is approximated by the cell gradient $\nabla T_{i,j}$. However, a modified

186 gradient $\nabla T_{u,v}$ calculated by the bilinear interpolation can be used to improve the path precision

187 (see *Fig. 3*).

Proposed algorithm

189 To improve the computational efficiency of global path planning in a large-scale and multi-

190 island marine environment, the proposed algorithm performs path planning twice for different

191 purposes. First, the path planning is performed in a low SR (LSR) grid map to determine an

192 effective region. The final global path is then obtained in the second path planning within the

193 effective region of a high SR (HSR) grid map. The relevant methods and procedures applied in

194 the algorithm are as follows. The complete algorithm flow is summarized in the last part of this

195 section.

Mapping of two-level SR grid maps

197 The two-level SR grid maps are contained in the HSR and LSR grid maps. The HSR grid map is

198 directly obtained from the Google satellite images data. A mapping relationship between the

199 LSR cells and the corresponding $L \times L$ HSR cell sub-blocks is established, which can be

200 expressed as:

201 $\quad (i_L, j_L) \sim \begin{bmatrix} \left(i_H', j_H' + L - 1\right) & \cdots & \left(i_H' + L - 1, j_H' + L - 1\right) \\ \vdots & \ddots & \vdots \\ \left(i_H', j_H'\right) & \cdots & \left(i_H' + L - 1, j_H'\right) \end{bmatrix}$ (6)

202 where (i_L, j_L) are the LSR cell coordinates, and $\left(i_H', j_H'\right)$ are the original cell coordinates of the

203 mapped HSR cell sub-block, as shown in *Fig. 4*.

204 $\quad$ To map the LSR grid map to a sub-block of the HSR grid map, the following is used:

205 $\quad \begin{cases} i_H' = i_{Ho} + L i_L \\ j_H' = j_{Ho} + L j_L \end{cases}$ (7)

206 $\quad$ To map from a sub-block of the HSR grid map to a cell of the LSR grid map, we use:

207 $\quad \begin{cases} i_L = f_{\mathrm{floor}}\left(\frac{i_H' - i_{Ho}}{L}\right) \\ j_L = f_{\mathrm{floor}}\left(\frac{j_H' - j_{Ho}}{L}\right) \end{cases}$ (8)

208   where $(i_H, j_H), i'_H \leq i_H \leq i'_H + L - 1, j'_H \leq j_H \leq j'_H + L - 1$ are the HSR cell coordinates. (i_{Ho}, j_{Ho})

209   are the original cell coordinates of the original HSR cell sub-block (see *Fig. 4*), which are

210   determined by:

211
$$\begin{cases} i_{Ho} = \left(i_{Hg} - f_{floor}\left(\frac{L}{2}\right)\right)\%L \\ j_{Ho} = \left(j_{Hg} - f_{floor}\left(\frac{L}{2}\right)\right)\%L \end{cases} \tag{9}$$

212   where (i_{Hg}, j_{Hg}) are the HSR cell coordinates of the goal cell x_g, the function $f_{floor}(m)$ indicates

213   a rounding down of m, and $a\%b$ indicates the modulo operation.

214   The SR of the LSR grid map is:

215    $D_{LRes} = LD_{HRes}$ (10)

216   where D_{HRes} is the SR of the HSR grid map. In general, L is defined as:

217    $L \leq f_{round}\left(\frac{D_{Th}}{2D_{HRes}}\right)$ (11)

218   where D_{Th} is the distance threshold of the virtual obstacle influence, and the function $f_{round}(m)$

219   indicates a rounding of m. The cell numbers of the LSR grid map in the X-axis and Y-axis

220   directions are:

221    $M_L = f_{floor}\left(\frac{M_H - i_{Ho}}{L}\right)$ (12)

222    $N_L = f_{floor}\left(\frac{N_H - j_{Ho}}{L}\right)$ (13)

223   where M_H and N_H are the cell numbers of the HSR grid map in the X-axis and Y-axis directions,

224   respectively.

225   Based on the mapping relationship, an LSR cell type is determined by comparing the setting

226   threshold Γ and obstacle cell proportion γ in the HSR cell sub-block. If $\gamma > \Gamma$, the LSR cell is set

227   as an obstacle cell; otherwise, it is set as a free cell. Γ prefers to select a small value to retain as

228   much obstacle information as possible.

IDC-FM² method

230   In the basic FM² method, the path is improved using a modified speed map. Two parameters, α

231   and β, are used to modify the speed map. These modifications ensure the flexibility of the

232   computed path. However, these two parameters have no clear physical meaning. The

233   normalization of the speed values in the FM-1st step also results in several shortcomings. For

234   example, all the speed values in the speed map must be calculated. When the map range changes

235   slightly, the entire speed map rescales and differs in the same locations of the original map, and

236   the parameters must be adjusted.

237   An IDC-FM^2 method is proposed to address these shortcomings. In this method, a time-cost
238   weighting function, w_x, is introduced to adjust the speed map. Three inshore-distance parameters
239   (the distance threshold D_{Th}, as well as the virtual strong and weak constraint distance parameters,
240   D_{sc} and D_{wc}, respectively) and two weighting function values with respect to D_{sc} and D_{wc} (w_{sc}
241   and w_{wc}, respectively) are used to determine the function w_x, and D_{Th} is used to achieve the
242   same function as the speed saturation. Although five parameters are set, only two inshore-
243   distance parameters, D_{Th} and D_{sc}, must be considered, while suitable values of w_{sc} and w_{wc} can
244   be selected as constant values. Based on this method, the path portion around an obstacle is
245   constrained in a region determined by D_{Th} and D_{sc}.

246   *Time-cost weighting function*

247   The time-cost weighting function w_x is introduced as a concept of the virtual obstacle influence.
248   When a USV is far away from obstacles, there is no obstacle influence. As the USV approaches,
249   the virtual obstacle influence increases slowly at first and the amplitude increases gradually. The
250   obstacle influence increases dramatically when the USV is close to within a certain degree.
251   Similar to the FM-1st step in the basic FM^2 method, all obstacle cells are set as source cells to
252   first calculate the arrival time map. The difference here is that a threshold $T_{Th} = \tau^{D_{Th}}/D_{Res}$
253   could be set to speed up the calculation, where D_{Res} is the SR of the map, and τ is the unified
254   time-cost which can take 1 as the value. In the loop procedure, if the smallest T value of a cell in
255    $\mathbf{S}_T$ is larger than or equal to T_{Th}, all T values of non-accepted cells can be set as T_{Th} directly.
256   Based on the resulting arrival time map, $\mathbf{T}_{sat}$, with a saturation threshold, the time-cost
257   weighting function map, $\mathbf{W}'$, is designed as:

$$258 \quad w_x' = \begin{cases} \infty & ,T_x = 0 \\ 1 + a\left(\frac{T_{Th}}{T_x} - 1\right)^b & ,T_x > 0 \end{cases} \tag{14}$$

259   where w_x' and T_x are the values of the cell x in $\mathbf{W}'$ and $\mathbf{T}_{sat}$, respectively, and a and b are two
260   positive coefficients to be determined.
261   When a free grid, x, is within the virtual influenced scope of obstacles (i.e., $0 < D_x \le D_{Th}$,
262   where D_x is the closest distance to obstacles), then:

$$263 \quad D_x \approx D_{Res}\frac{T_x}{\tau} \tag{15}$$

264   Therefore, w_x with respect to D_x is used to approximate w_x':

$$265 \quad w_x(D_x) = \begin{cases} \infty & ,D_x = 0 \\ 1 + a\left(\frac{D_{Th}}{D_x} - 1\right)^b & ,0 < D_x \le D_{Th} \\ 1 & ,D_x > D_{Th} \end{cases} \tag{16}$$

266   where $w_x(D_x)$ is the value in cell x of the approximate time-cost weighting function map, $\boldsymbol{W}$.

267   The time-cost value of each grid in the time-cost map, which is the inversion of the speed map, is

268   then adjusted to:

269    $new\tau_x = w_x(D_x)\tau_x$ (17)

270   Eq. (16) implies that w_x increases when grid x is closer to obstacles, and it also increases

271   when D_x is small. The planned path tends to select points with a lower arrival time-cost.

272   Therefore, when the path is close to obstacles, it will be farther away from obstacles compared to

273   the path planned by the basic FMM.

274   ***Parameters for the time-cost weighting function***

275   Five parameters, D_{Th}, D_{sc}, D_{wc}, w_{sc}, and w_{wc}, are introduced to determine the coefficients a and

276   b, which are determined by:

277    $b = \dfrac{\ln(w_{\mathrm{sc}} - 1) - \ln(w_{\mathrm{wc}} - 1)}{\ln(1 - e_{\mathrm{sc}}) - \ln(1 - e_{\mathrm{wc}}) + \ln e_{\mathrm{wc}} - \ln e_{\mathrm{sc}}}$ (18)

278    $a = (w_{\mathrm{sc}} - 1)\left(\dfrac{e_{\mathrm{sc}}}{1 - e_{\mathrm{sc}}}\right)^b$ (19)

279   where $e_{\mathrm{sc}} = D_{\mathrm{sc}}/D_{\mathrm{Th}}$, $e_{\mathrm{wc}} = D_{\mathrm{wc}}/D_{\mathrm{Th}}$, $D_{\mathrm{sc}} < D_{\mathrm{wc}} < D_{\mathrm{Th}}$, and $w_{\mathrm{sc}} > w_{\mathrm{wc}} > 1$.

280   The parameter D_{Th} determines the largest virtual influence scope of the obstacles. The

281   suggestion for the selection of D_{Th} is that there should be a sufficiently safe buffer area for

282   obstacles. The parameter D_{sc} is very important for the safety of the USV. First, it is suggested

283   that this parameter satisfies:

284    $D_{\mathrm{sc}} \geq v_{\mathrm{U,max}}t_{\mathrm{r}} - \dfrac{v_{\mathrm{U,max}}^2}{2a_{\mathrm{d}}}$ (20)

285   where $v_{\mathrm{U,max}}$ is the maximum speed of the USV, t_{r} is the reaction time of the thruster, and a_{d} is

286   the negative maximum acceleration of the USV under braking. In an unknown environment, it is

287   better to select a slightly larger value to ensure safety, which is similar in environments with

288   shallow and reef waters. This value may be relatively small in deep and reef-free waters to

289   improve path efficiency. D_{wc} is a parameter that acts as the adjusted constraint distance. In a

290   non-channel ocean area, the path portion around the obstacles will be limited to the region

291   between D_{wc} and D_{Th}. The D_{wc} value can be set independently or determined jointly by D_{Th} and

292   D_{sc}. In this study, it is set as:

293    $D_{\mathrm{wc}} = D_{\mathrm{Th}} - \dfrac{\sqrt{2}}{2}(D_{\mathrm{Th}} - D_{\mathrm{sc}})$ (21)

294   The area around the obstacles is divided into four parts by the three inshore-distance

295   parameters: D_{sc}, D_{wc}, and D_{Th}. The three parts extending from an obstacle to the outside region

296   are the danger region, S_{d}; the strong constraint region, S_{sc}; and the weak constraint region, S_{wc}

297   (see *Fig. 5*). The influences of w_{wc} and w_{sc} on the path are shown in *Fig. 5*. As shown in *Fig.*

298   *5A*, the path will be located far from the obstacle until it is near the boundary of S_{wc} when w_{wc} is

299   large. In contrast, the path is closer to an obstacle when w_{sc} is large (see *Fig. 5B*). The minimum

300   degree of the path close to S_{sc} is mainly determined by w_{wc}, which can be inferred by further

301   comparing the paths in *Fig. 5B* with the closest path to the obstacle ($w_{wc} = 1.1$) in *Fig. 5A*.

302   However, regardless of the change in w_{wc} and w_{sc} ($w_{sc} > w_{wc} > 1$), the path portions bypassing

303   obstacles will always be within S_{wc} or around the outside boundary of S_{wc} in non-channel ocean

304   areas. In channel areas where there is no S_{wc}, the path will follow the quasi-centerline of the

305   channel if it passes through the channel. In this study, the values of $w_{sc} = 40.0$ and $w_{wc} = 2.0$

306   were selected and fixed. The paths can be flexibly modified by the inshore-distance parameters

307   D_{Th} and D_{sc}.

Determination of effective regions within the HSR grid map

309   The effective region within the HSR grid map is important for improving the computational

310   efficiency of the proposed algorithm. It is acquired based on the mapping between the LSR and

311   HSR grid maps when the corresponding region within the LSR grid map is determined.

312   Two effective regions that are respectively used in the FM-1st and FM-2nd steps of the IDC-

313   FM^2 method must be determined. The low-precision initial path, ℓ_{ini}, is obtained first based on

314   the LSR arrival time map $\mathcal{T}_L$. Two effective regions within the LSR grid map, S_{LER_1st} and

315   S_{LER_2nd}, are determined by expanding the grids around the "path-passed" grids of ℓ_{ini}. As shown

316   in *Fig. 6*, the nearest neighbor grid among the four neighbor cells of the waypoint $P_i = (x_i, y_i)$ is

317   defined as the "path-passed" grid. When all "path-passed" grids are determined, the $\kappa_{2nd} = \kappa$

318   levels of the cells are extended to obtain S_{LER_2nd}. Considering S_{LER_1st}, there are three different

319   situations:

320   •   ***Situation*** 1: there is no cell with $w_x > 1$ in S_{LER_2nd};

321   •   ***Situation*** 2: there is at least one cell with $w_x > 1$ and no obstacle cell in S_{LER_2nd}.

322   •   ***Situation*** 3: there is at least one obstacle cell in S_{LER_2nd}.

323   In ***Situations*** 1 and 3, $\kappa_{1st} = \kappa$. In ***Situation*** 2, S_{LER_2nd} is extended continuously with $\Delta\kappa$

324   levels of cells until there is at least one obstacle cell for the first time, and $\kappa_{1st} = \kappa + \Delta\kappa$. Finally,

325   two corresponding effective regions within the HSR grid map, S_{HER_1st} and S_{HER_2nd}, are

326   determined and used in the different FM steps of the IDC-FM^2 method to obtain the HSR arrival

327   time map $\mathcal{T}_H$.

328   The extended parameter κ affects the paths (recorded as ℓ_κ) based on the proposed algorithm.

329   When κ is not adequately large, these paths may not be consistent with the reference path ℓ_{ref},

330   which is acquired by applying the IDC-FM^2 method in the HSR grid map directly. A navigable

331   global path is used as a case to compare the consistency between ℓ_κ and ℓ_{ref}. The basic

332   parameters used in the proposed algorithm are listed in *Table 1*. Using ℓ_{ref} as the reference, the

333   distances between the corresponding waypoints of ℓ_κ ($\kappa = 3,\ldots,6$) and ℓ_{ref} are shown in *Fig. 7*. It

334   is shown that the largest distance decreases when κ increases, and the distance becomes 0 when

335    $\kappa \geq 7$. Other cases were tested and similar results were obtained. These results indicate that there

336   is good consistency when κ is sufficiently large (such as $\kappa = 10$). $\Delta\kappa$ is a variable parameter that

337   was determined based on three different cases.

338   **Algorithm flow**

339   The proposed algorithm is improved based on the basic FM²-based algorithm. The basic FM²-

340   based algorithm is executed directly on a single grid map. The algorithm flow of this basic

341   algorithm is shown in *Fig. 8* in the form of main data stream and used methods. This algorithm

342   flow has two main steps. In *Step* S1, an arrival time map with obstacle sources and a saturation

343   threshold, $\mathcal{T}_{sat}$, is obtained by first applying the FMM. Then, the approximate time-cost

344   weighting function map, W, is calculated and used to adjust the time-cost map. Based on the

345   adjusted time-cost map, an arrival time map with the goal point source, $\mathcal{T}$, is obtained by

346   applying the FMM again. The entire process is used to obtain $\mathcal{T}$ by applying the IDC-FM²

347   method. Finally, the global path is acquired based on $\mathcal{T}$ by applying the gradient descent method

348   in *Step* S2.

349   The main data stream in the proposed algorithm and the methods used to obtain them are

350   shown in *Fig. 9*. The main flow is as follows:

351   1.   *Step* T1. Obtain the two-level SR grid maps;

352   2.   *Step* T2. Obtain the LSR arrival time map $\mathcal{T}_L$ in the LSR grid map;

353   3.   *Step* T3. Determine the effective regions within the HSR grid map (S_{HER_1st} and S_{HER_2nd});

354   4.   *Step* T4. Obtain the HSR grid arrival time map $\mathcal{T}_H$ with the effective region constraint in

355   both FM steps of the IDC-FM² method. There are several adjustments in both FM steps

356   compared to the basic FMM performance. For the FM-1st step, all w_x values within S_{HER_1st}

357   need not be adjusted, and this step can be ignored in ***Situation*** 1. In ***Situation*** 2 and

358   ***Situation*** 3, all free cells out of S_{HER_1st} are set to $\boldsymbol{S}_A$ with $T = \infty$ in the initialization of this

359   step. In the FM-2nd step, all free cells out of S_{HER_2nd} are set to $\boldsymbol{S}_A$ with $T = \infty$ in the

360   initialization;

361   5.   *Step* T5. Acquire the final high precision global path by applying the gradient descent

362   method based on $\mathcal{T}_H$.

363   # Results and discussion

364   **Simulation environments**

365   Two surrounding spatial areas of Zhucha Island and Changhai County were selected as the

366   simulation environments. The original maps adopted Google satellite images, as shown in *Fig.*

367   *10*. The spatial resolutions in the longitude and latitude directions are approximately 4.8 m and

368 3.9 m, respectively. Temporary binary grid maps are obtained based on the image processing
369 method (*Shi et al., 2018*) combined with manual assistance, and then the HSR grid maps with
370 $D_{\mathrm{HRes}} = 10$ m in both the longitude and latitude directions are determined by resampling
371 processing. The corresponding binary grid maps are presented in *Fig. 11*. Their ranges are
372 7km × 7km and 64km × 48km, respectively.

Inshore-distance-constraint performances

374 A path planning from $P_{\mathrm{s}} = [4.3$ km, 2.8 km$]$ to $P_{\mathrm{g}} = [3.6$ km, 2.0 km$]$ is used as a typical case to
375 analyze the inshore-distance-constraint performance of the IDC-FM² method using the inshore-
376 distance parameters D_{Th} and D_{sc}. As shown in *Fig. 12*, the path planned based on the basic
377 FMM, ℓ_{FMM}, is very close to the islands when this path bypasses them. For comparison, all paths
378 planned by the proposed algorithm, ℓ_1 to ℓ_4, are located away from islands by a certain distance.
379 They are therefore significantly better choices from a safety perspective.
380   Paths ℓ_1 to ℓ_4 are acquired using different inshore-constraint distance parameters (D_{Th} and D_{sc}
381 , see *Table 2* wherein D_{wc} is calculated based on D_{Th} and D_{sc}) in the IDC-FM² method in which
382 $w_{\mathrm{sc}} = 40.0$ and $w_{\mathrm{wc}} = 2.0$. These paths are clearly adjusted by these distance parameters. When
383 the distance constraints (D_{Th} and D_{sc}) are small, the path (such as ℓ_1, see *Fig. 12*) will be a
384 somewhat close to the islands. The estimated quasi-closest distance from ℓ_1 to an island is
385 approximately 38 m, which is shorter than half of the shortest channel width ($d_{\mathrm{hscw}} \approx 101$m). In
386 this situation, ℓ_1 is always outside of its S_{sc} (<28.2 m) value, and the path portions around the
387 islands are located in the region of its S_{wc} (28.2 m to 60.0 m). When the distance constraints
388 increase, the paths (such as ℓ_2 and ℓ_3, see *Fig. 12*) in the non-channel areas will be outside of S_{sc}.
389 The estimated quasi-closest distances (approximately 115 m and 128 m, respectively) were
390 larger than d_{hscw}. However, ℓ_2 and ℓ_3 will be along the quasi-midline of the channel in the
391 channel area, regardless of whether d_{hscw} is greater or less than D_{wc} for these two paths because
392 the weighting time-cost will be smaller than the path around the islands (as in ℓ_4, see *Fig. 12*). If
393 the distance constraints are further increased, ℓ_4 will be selected for safety, although more time
394 will be required. These results indicate that the path based on the proposed algorithm can be
395 adjusted by setting different inshore-constraint distance parameters to meet safety requirements.
396 Additionally, when the inshore-constraint distance parameters are determined, the path will cross
397 a channel when its width is adequately large; otherwise, the path will bypass around the islands.
398   One classical approach to plan collision-free paths is applying the FMM within the map with
399 inflated obstacles. For comparison, paths planned by the classical approach and the IDC-FM²
400 method are shown in *Fig. 13*. The starting and goal positions are $P_{\mathrm{s}} = [4.15$ km, 2.74 km$]$ and P_{g}
401 $= [4.33$ km, 2.40 km$]$, and the island obstacles have been inflated by 94.0 m. As shown in *Fig.*
402 *13*, the path planned by the classical approach, $\ell_{\mathrm{FMM + inflated}}$, is obviously influenced by the

403   shape of the inflated obstacle. In some path turns which are marked by ellipses in *Fig. 13*, they
404   are both affected by the shape of the inflated obstacle and the time-value gradient calculation of
405   cells adjacent to the inflated obstacle. A smooth path, $\ell_{\text{FMM + inflated + smooth}}$, is also shown in *Fig.*
406   *13*. Although it removes abrupt turn waypoints, the path is still not very smooth. As
407   comparisons, paths planed based on the IDC-FM^2 method, ℓ_1 and ℓ_2, are obviously smoother
408   than $\ell_{\text{FMM + inflated + smooth}}$. In addition, when using the proposed rapid path planning algorithm
409   based on two-level SR grid maps to improve the computational efficiency, the channel will be
410   mapped as obstacle cells when mapping the HSR grid map to the LSR grid map in this case
411   (taking the mapping parameter $L = 8$ as shown in *Table 1*), because of the influence of the
412   inflated obstacle. This unexpected mapping will result in the loss of the path through the channel.

Global path planning in a large-scale and complex multi-island environment

414   To verify the path planning ability and the computational efficiency improvement of the
415   proposed algorithm in a large-scale and relatively complex multi-island environment, the
416   simulation environment shown in *Fig. 11B* is selected, and the long-length path cases of USVs
417   around the islands or across channels are investigated.

Path planning cases

419   Five typical path cases were selected to demonstrate the performance of the proposed algorithm.
420   The P_s and P_g groups are listed in *Table 3*. When determining the inshore-distance parameter D_{sc}
421   , a USV with $v_{\text{U,max}} = 6$ m/s, $t_r = 2$ s, $a_d = -1$ m/s^2 is considered as a case. Based on Eq. (20),
422   D_{sc} is suggested to select a value larger than 30 m. Therefore, a larger value $D_{\text{sc}} = 50$ m is used
423   for the path planning for the selected five path cases. Another inshore-distance parameter D_{Th}
424     $= 200$ m is selected empirically. The main parameters used in these path planning cases are
425   shown in *Table 1*.
426   As shown in *Fig. 14*, all paths successfully bypass the islands. The enlarged views (see *Fig.*
427   *15*, every path is displayed with a solid line) provide further details. Every path maintains a
428   relatively safe distance when close to an island, and smooth when turning around the island. For
429   a narrow channel of a certain degree (usually wider than $2D_{\text{LRes}}$), the path is planned along the
430   quasi-midline of the channel to ensure that it is as safety as possible. The path lengths range from
431   about 26.52 km to 47.86 km (see *Table 3*). These lengths can cover the range of most
432   applications of current small-and medium-sized USVs. These results show the effective path
433   planning ability of the proposed algorithm in a large-scale and complex multi-island
434   environment.

Computational efficiency improvement

436   To verify the computational efficiency improvement of the proposed method, the time spent on
437   the five typical paths (see *Table 3*) based on the proposed algorithm was calculated. For
438   comparison, the time spent on the basic algorithm which is executed directly on a single HSR
439   grid map by applying the IDC-FM^2 method was also calculated as a reference. The C algorithm

440   code was tested on a computer with a Core i5-6300U CPU and 8G memory, which runs a 64-bit
441   Win7 operating system.
442   Path planning for every typical path was repeated 10 times. Every planning time starts from
443   the HSR grid map reading (in *Steps* S1 and T1) and ends when the global path has been
444   calculated (in *Steps* S2 and T5). Because the path planning is performed on a Windows operating
445   system, which involves multitasking, there may be many factors influencing the planning time,
446   such as the variable CPU usage, random CPU hit rate to the cache, and possible thread
447   scheduling. Therefore, the average planning time is used to evaluate the computational
448   efficiency. The average planning time results of the five planned paths are presented in *Table 4*.
449   These results indicate that the computational efficiency of the proposed algorithm based on two-
450   level grid maps is significantly higher than the algorithm based on a single HSR grid map. The
451   time for all cases is approximately 2 s, indicating that this method can be used in less demanding
452   real-time planning applications. This can effectively improve the practicality of the proposed
453   algorithm.
454   When planning a path by applying the basic FM^2-based algorithm, if the grid map scale is very
455   large, two aspects of calculations will severely increase compared with small-scale maps. First,
456   as the number of cells increases significantly, the calculations for free cells increase. Second, $\boldsymbol{S}_\text{T}$
457   changes in every iteration of calculations. The overall scale of $\boldsymbol{S}_\text{T}$ clearly increases, resulting in
458   the increase in sorting time of the adopted priority heap struct. This compared algorithm, whose
459   algorithm flow is shown in *Fig. 8*, is used directly on a single HSR grid map. The valid scale of
460   the HSR grid map is the number of free cells. For comparison, the proposed algorithm
461   determines the effective regions within the HSR grid map first and then plans a path within the
462   determined effective regions. As shown in *Fig. 9*, *Steps* T1–T3 complete the determination of
463   effective regions, and *Steps* T4–T5 realize the path planning. The same function of the path
464   planning is also realized by *Steps* S1–S2, as shown in *Fig. 8*. Comparing the path planning steps
465   of the compared and proposed algorithms, the numbers of free cells in effective regions are much
466   less than the ones in the HSR grid map. This indicates that the calculations for free cells and the
467   overall scale of $\boldsymbol{S}_\text{T}$ will greatly decrease. Therefore, the planning time of the proposed algorithm
468   reduces. Certainly, there are more steps in the proposed algorithm. Planning time spent on these
469   steps (*Steps* T1–T3) mainly depends on the LSR mapping parameter L. When the planning time
470   spent on the compared algorithm, t_ref, is large enough, the planning time spent on *Step* T2 can
471   reduce to less than t_ref/L^2 (in the cases shown in *Table 4*, this value is approximately $t_\text{ref}/(2L^2)$).
472   The planning time spent on *Steps* T1 and T3 is often much less than t_ref. Therefore, the increased
473   planning time spent on *Steps* T1–T3 is much less than the saved planning time spent on *Steps*
474   T4–T5.

475   # Conclusions

476 In some special USV applications such as sea rescue, it requires the USV to reach a goal position
477 as soon as possible. FMM is a suitable global path planning method which can plan a time-
478 optimal global path. However, the planned paths are not safe enough when they bypass
479 obstacles, because of that they are too close to the obstacles. Therefore, FMM should be
480 improved for safety considerations. One classical approach is applying the FMM within the map
481 with inflated obstacles. Although the planned paths are safe by inflating the obstacle size, there
482 may be abrupt turns when the obstacles have sharp corners, and the planned paths are influenced
483 by shapes of inflated obstacles and may be not very smooth. Such paths are usually unfriendly
484 for USVs. A rapid global path planning algorithm applying an IDC-FM^2 method in two-level SR
485 grid maps is proposed for USV applications with short time requirements in large-scale and
486 complex multi-island environments. This algorithm can acquire a continuous, smooth, quasi-
487 time-optimal path while maintaining a safe distance around obstacles when bypassing obstacles.
488 When the path is near obstacles, it is limited to a safe area determined by two inshore-distance
489 parameters. By adjusting these two inshore-distance parameters, the path can be modified
490 flexibly. Although the time optimality is missed by using the IDC-FM^2 method, the safety with a
491 higher priority in most applications has been ensured initially while the optimal loss of time
492 remains at a certain level. The two-path planning process based on two-level SR grid maps
493 improves the computational efficiency compared with the basic FM^2-based method. The
494 planning time on the order of seconds is acceptable in many global path planning applications of
495 USVs. Meanwhile, the planning time of this order of magnitude is typically short enough for
496 USVs in most situations with the requirement to replan the path. This indicates the potential of
497 replanning by using the proposed algorithm from the perspective of planning time. As a
498 comparison, when using the classical approach which applies the FMM within the map with
499 inflated obstacles, narrow channels will be easier to map as obstacles when mapping the HSR
500 grid map to the LSR grid map in the rapid planning process based on two-level SR grid maps.
501 This shortcoming may result in the loss of some possible paths through channels.
502   On the other hand, there are still some challenges when using the proposed algorithm. For
503 example, how to accurately obtain the location information of newly detected obstacles and how
504 to add this information into the grid map when replanning are the common challenges. The IDC-
505 FM^2 method also needs to be modified to plan the path meeting the dynamic characteristics of a
506 USV. Otherwise, the USV may not follow the replanned path successfully at the beginning of the
507 path.
508   One shortcoming of the proposed algorithm is that paths through channels that are very narrow
509 but still passable in reality may be missed because of the first path planning process in the LSR
510 grid map. The introduction of environmental effects into the proposed algorithm is an important
511 task to be performed in future works. Marine experiments should also be conducted to verify the
512 validity of the algorithm.
513

Acknowledgments

515 We would like to thank Editage (www.editage.cn) for English language editing.

516

517

518

519

520

521

522

References

524 Adalsteinsson D, and Sethian JA. 1995. A fast level set method for propagating interfaces. *Journal of*
525   *Computational Physics* 118:269-277. 10.1006/jcph.1995.1098
526 Alvarez D, Gómez J, Garrido S, and Moreno L. 2015. 3D robot formations path planning with fast
527   marching square. *Journal of Intelligent & Robotic Systems* 80:507-523. 10.1007/s10846-015-
528   0187-1
529 Amorim D, and Ventura R. 2014. Towards efficient path planning of a mobile robot on rough terrain.
530   2014 IEEE International Conference on Autonomous Robot Systems and Competitions
531   (ICARSC). Espinho, Portugal: IEEE. p 22-27.
532 Arzamendia M, Gregor D, Reina DG, and Toral SL. 2019. An evolutionary approach to constrained path
533   planning of an autonomous surface vehicle for maximizing the covered area of Ypacarai Lake.
534   *Soft Computing* 23:1723-1734. 10.1007/s00500-017-2895-x
535 Campbell S, Naeem W, and Irwin GW. 2012. A review on improving the autonomy of unmanned surface
536   vehicles through intelligent collision avoidance manoeuvres. *Annual Reviews in Control* 36:267-
537   283. 10.1016/j.arcontrol.2012.09.008
538 Dijkstra EW. 1959. Communication with an automatic computer doctor PhD thesis. University of
539   Amsterdam.
540 Gómez JV, Alvarez D, Garrido S, and Moreno L. 2019. Fast methods for Eikonal equations: an
541   experimental survey. *IEEE Access* 7:39005-39029. 10.1109/ACCESS.2019.2906782
542 Gómez JV, Lumbier A, Garrido S, and Moreno L. 2013. Planning robot formations with fast marching
543   square including uncertainty conditions. *Robotics and Autonomous Systems* 61:137-152.
544   https://doi.org/10.1016/j.robot.2012.10.009
545 Garrido S, Alvarez D, and Moreno LE. 2020. Marine applications of the fast marching method. *Frontiers*
546   *in Robotics and AI* 7:1-12. 10.3389/frobt.2020.00002
547 Garrido S, Moreno L, Blanco D, and Martin F. 2007. FM2: A real-time fast marching sensor-based
548   motion planner. 2007 IEEE/ASME international conference on advanced intelligent
549   mechatronics. Zurich, Switzerland: IEEE. p 1-6.
550 González V, Monje CA, Moreno L, and Balaguer C. 2016. Fast marching square method for UAVs
551   mission planning with consideration of Dubins model constraints. *IFAC-PapersOnLine* 49.
552   10.1016/j.ifacol.2016.09.029
553 Guo J, Gao Y, and Cui G. 2013. Path planning of mobile robot based on improved potential field.
554   *Information Technology Journal* 12:2188-2194. 10.3923/itj.2013.2188.2194
555 Hart PE, Nilsson NJ, and Raphael B. 1968. A formal basis for the heuristic determination of minimum
556   cost paths. *IEEE Transactions on Systems Science and Cybernetics* 4:100-107.
557   10.1109/TSSC.1968.300136
558 Khatib O. 1986. Real-time obstacle avoidance for robot manipulator and mobile robots. *The International*
559   *Journal of Robotics Research* 5:90-98. 10.1177/027836498600500106
560 Kim H, Kim D, Shin JU, Kim H, and Myung H. 2014. Angular rate-constrained path planning algorithm
561   for unmanned surface vehicles. *Ocean Engineering* 84:37-44. 10.1016/j.oceaneng.2014.03.034

562 Kim H, Kim SH, Jeon M, Kim J, Song S, and Paik KJ. 2017. A study on path optimization method of an
563   unmanned surface vehicle under environmental loads using genetic algorithm. *Ocean*
564   *Engineering* 142:616-624. 10.1016/j.oceaneng.2017.07.040
565 Lee T, Kim H, Chung H, Bang Y, and Myung H. 2015. Energy efficient path planning for a marine
566   surface vehicle considering heading angle. *Ocean Engineering* 107:118-131.
567   10.1016/j.oceaneng.2015.07.030
568 Lin Q. 2003. Enhancement, extraction, and visualization of 3D volume data doctor PhD thesis. Linköping
569   University.
570 Liu Y, and Bucknall R. 2015. Path planning algorithm for unmanned surface vehicle formations in a
571   practical maritime environment. *Ocean Engineering* 97:126-144.
572   10.1016/j.oceaneng.2015.01.008
573 Liu Y, and Bucknall R. 2016. The angle guidance path planning algorithms for unmanned surface vehicle
574   formations by using the fast marching method. *Applied Ocean Research* 59:327-344.
575   10.1016/j.apor.2016.06.013
576 Liu Y, Bucknall R, and Zhang X. 2017. The fast marching method based intelligent navigation of an
577   unmanned surface vehicle. *Ocean Engineering* 142:363-376. 10.1016/j.oceaneng.2017.07.021
578 Liu Z, Zhang Y, Yu X, and Yuan C. 2016. Unmanned surface vehicles: an overview of developments and
579   challenges. *Annual Reviews in Control* 41:71-93. 10.1016/j.arcontrol.2016.04.018
580 Messias J, Ventura R, Lima P, Sequeira J, Alvito P, Marques C, and Carriço P. 2014. A robotic platform
581   for edutainment activities in a pediatric hospital. 2014 IEEE International Conference on
582   Autonomous Robot Systems and Competitions (ICARSC). Espinho, Portugal: IEEE. p 193-198.
583 Naeem W, Irwin GW, and Yang A. 2012. COLREGs-based collision avoidance strategies for unmanned
584   surface vehicles. *Mechatronics* 22:669-678. 10.1016/j.mechatronics.2011.09.012
585 Sethian JA. 1999. *Level set methods and fast marching methods*. Cambridge, UK Cambridge University
586   Press.
587 Shi B, Su Y, Zhang H, Liu J, and Wan L. 2018. Obstacles modeling method in cluttered environments
588   using satellite images and its application to path planning for USV. *International Journal of*
589   *Naval Architecture and Ocean Engineering* 11:202-210. 10.1016/j.ijnaoe.2018.04.001
590 Singh Y, Sharma S, Sutton R, and Hatton D. 2018. Towards use of Dijkstra algorithm for optimal
591   navigation of an unmanned surface vehicle in a real-time marine environment with results from
592   artificial potential field. *TransNav the International Journal on Marine Navigation and Safety of*
593   *Sea Transportation* 12:125-131. 10.12716/1001.12.01.14
594 Song C. 2014. Global path planning method for USV system based on improved ant colony algorithm.
595   *Applied Mechanics and Materials* 568-570:785-788. 10.4028/www.scientific.net/AMM.568-
596   570.785
597 Song J, Hao C, and Su J. 2020. Path planning for unmanned surface vehicle based on predictive artificial
598   potential field. *International Journal of Advanced Robotic Systems* 17:1-13.
599   10.1177/1729881420918461
600 Song L, Mao Y, Xiang Z, Zhou Y, and Du K. 2015. A study on path planning algorithms based upon
601   particle swarm optimization. *Journal of Information & Computational Science* 12:673-680.
602   10.12733/jics20105310
603 Song R, Liu Y, and Bucknall R. 2017. A multi-layered fast marching method for unmanned surface
604   vehicle path planning in a time-variant maritime environment. *Ocean Engineering* 129:301-317.
605   10.1016/j.oceaneng.2016.11.009
606 Tan G, Zou J, Zhuang J, Wan L, Sun H, and Sun Z. 2020. Fast marching square method based intelligent
607   navigation of the unmanned surface vehicle swarm in restricted waters. *Applied Ocean Research*
608   95:102018. 10.1016/j.apor.2019.102018
609 Tsitsiklis JN. 1995. Efficient algorithms for globally optimal trajectories. *IEEE Transactions on*
610   *Automatic Control* 40:1528-1538.
611 Wang N, Jin X, and Er J. 2019. A multilayer path planner for a USV under complex marine
612   environments. *Ocean Engineering* 184:1-10. 10.1016/j.oceaneng.2019.05.017

613 Wang N, Xu H, Li C, and Yin J. 2020. Hierarchical path planning of unmanned surface vehicles: a fuzzy
614   artificial potential field approach. *International Journal of Fuzzy Systems*. 10.1007/s40815-020-
615   00912-y
616 Xia G, Han Z, Zhao B, Liu C, and Wang X. 2019. Global path planning for unmanned surface vehicle
617   based on improved quantum ant colony algorithm. *Math Probl Eng* 2019:1-10.
618   10.1155/2019/2902170
619 Xie S, Wu P, Liu H, Yan P, Li X, Luo J, and Li Q. 2016. A novel method of unmanned surface vehicle
620   autonomous cruise. *Industrial Robot: An International Journal* 43:121-130. 10.1108/IR-05-2015-
621   0097
622 Yan R, Pang S, Sun H, and Pang Y. 2010. Development and missions of unmanned surface vehicle.
623   *Journal of Marine Science and Application* 9:451-457. 10.1007/s11804-010-1033-2
624 Zhang W, Xu Y, and Xie J. 2019. Path planning of USV based on improved hybrid genetic algorithm.
625   2019 European Navigation Conference (ENC). Warsaw, Poland: IEEE. p 1-7.
626

Basic parameter values for the proposed algorithm.

1   **Table 1:**

2   **Basic parameter values for the proposed algorithm.**

Parameter	Value	Parameter	Value	Parameter	Value
L	8	Γ	0.2	D_{Th} (m)	200
D_{sc} (m)	50	w_{sc}	40.0	w_{wc}	2.0

Inshore-constraint distance parameters for the determination of w_x.

1   **Table 2:**

2   **Inshore-constraint distance parameters for the determination of w_x.**

Path	D_{Th} (m)	D_{wc} (m)	D_{sc} (m)
ℓ_1	60.0	28.2	15.0
ℓ_2	200.0	79.8	30.0
ℓ_3	200.0	104.5	65.0
ℓ_4	200.0	118.7	85.0

Quantified information of the five typical paths.

1   **Table 3:**

2   **Quantified information of the five typical paths.**

Path	P_s (km)	P_g (km)	Length (km)
ℓ_1	[35.34, 39.25]	[15.31, 11.65]	38.84
ℓ_2	[19.42, 41.02]	[17.10, 3.63]	37.90
ℓ_3	[42.96, 43.67]	[46.34, 8.24]	38.20
ℓ_4	[36.11, 18.77]	[47.44, 41.01]	26.47
ℓ_5	[3.95, 26.52]	[50.45, 30.83]	47.82

Planning times of five typical path cases for the compared algorithm based on a single HSR grid map and the proposed algorithm based on two-level SR grid maps.

1   **Table 4:**

2   **Planning times of five typical path cases for the compared algorithm based on a single HSR**

3   **grid map and the proposed algorithm based on two-level SR grid maps.**

Path	Average planning times (s)	
	Compared algorithm	**Proposed algorithm**
ℓ_1	29.21	1.94
ℓ_2	24.65	1.95
ℓ_3	30.25	1.97
ℓ_4	16.44	1.76
ℓ_5	40.40	2.24

Figure 1

Diagram of updating the neighbor of x_m in a 4-neighbor scheme.

Cell x is the neighbor cell to be updated. In this case, x is in $\mathbf{S}_\mathrm{T}$. If x is a cell in $\mathbf{S}_\mathrm{F}$, it moves from $\mathbf{S}_\mathrm{F}$ to $\mathbf{S}_\mathrm{T}$.
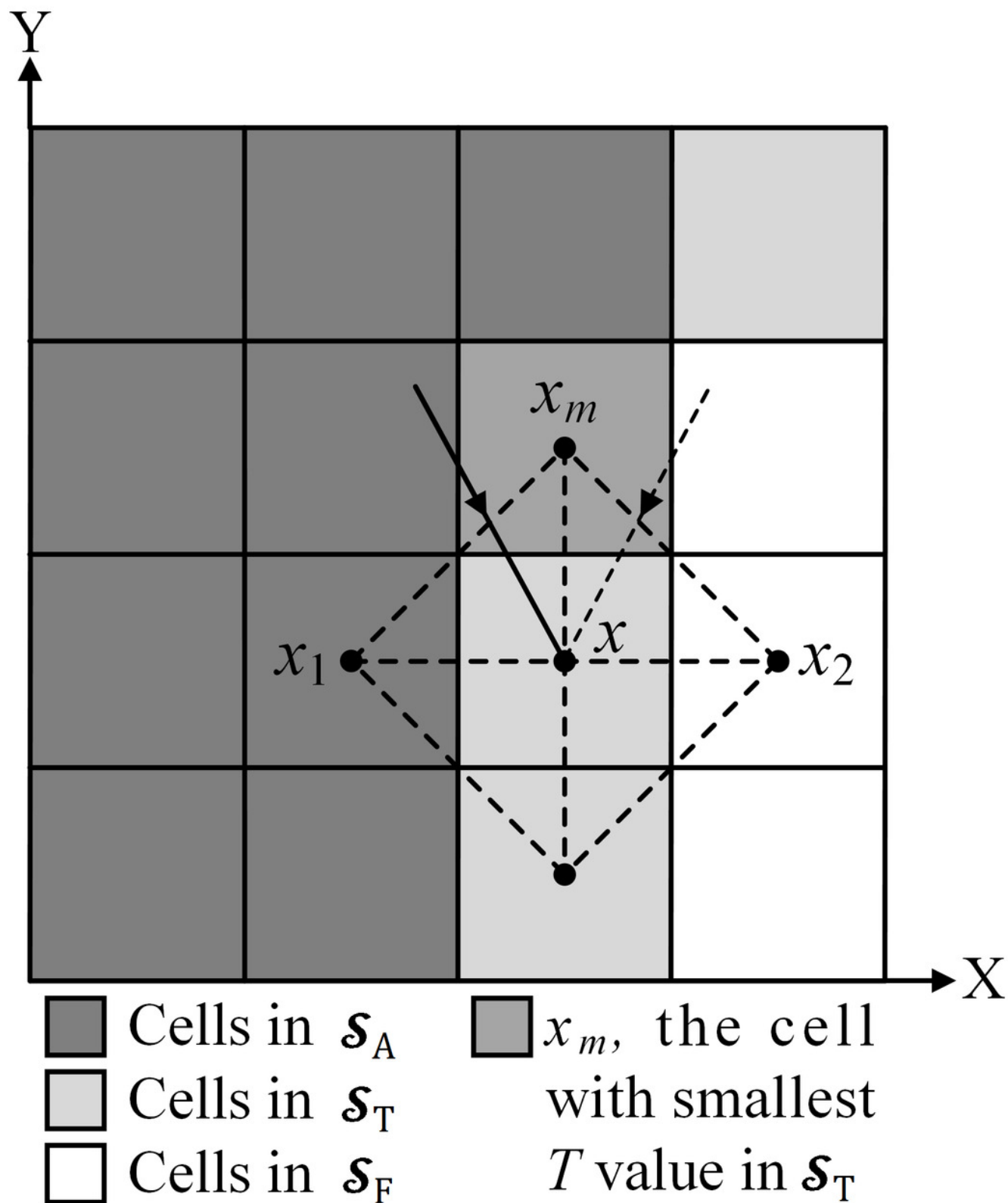
Figure 2

Maps used or generated in the process of using the FM^2 method.

(A) Binary grid map of an obstacle environment. The black cells are obstacle cells, and the white cells are free cells. (B) Arrival time map expanding from the obstacles, which is proportional to the distance from the obstacles. (C) Speed map with a saturated value. The black areas (0 value) represent the obstacles, while the white areas (1 value) represent the areas with velocity saturation. The other values are rescaled to better demonstrate the trend.

(D) Arrival time map obtained by applying the FM^2 method with x_g=(50,40). The path can be acquired by using the gradient descent method based on this map.
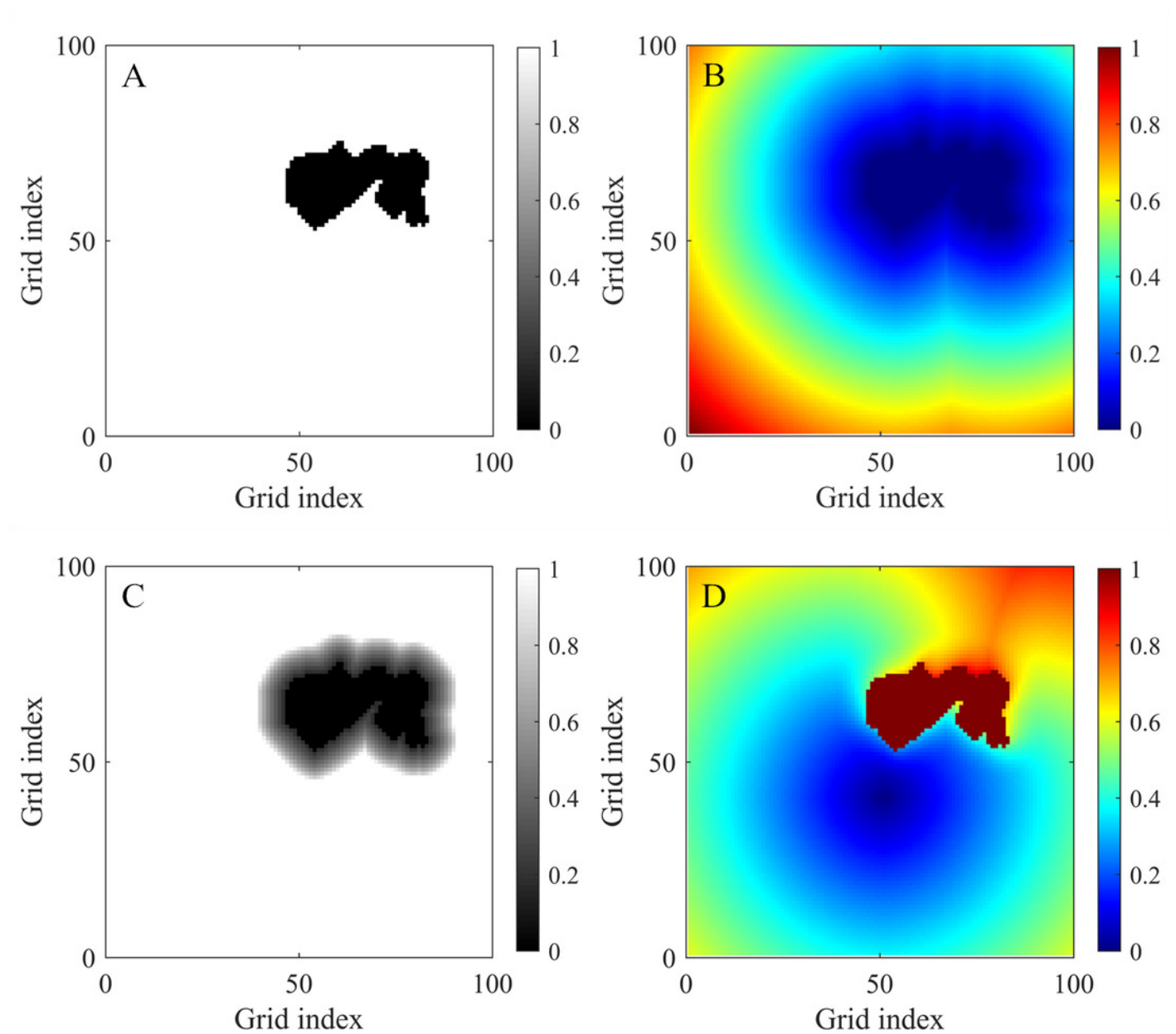
Figure 3

Modified gradient of a path waypoint calculated by the bilinear interpolation method.
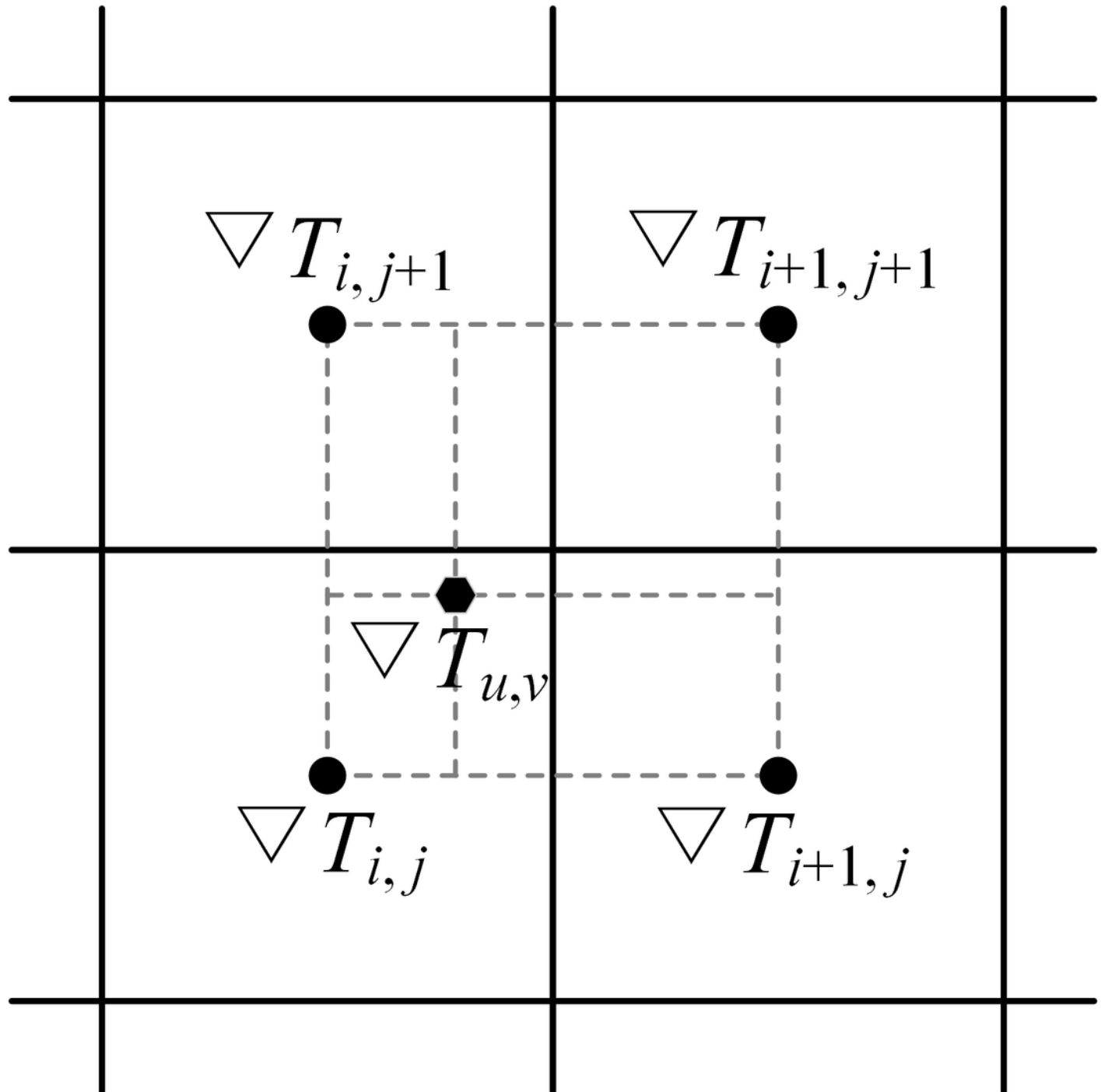
Figure 4

Mapping between an LSR grid map and an HSR grid map.

Here, a case with $L=4$ is shown. (i_{Ho}, j_{Ho}) is adaptively fine-tuned based on x_g. Therefore, x_g is set as the center (L is odd) or the quasi-center (L is even) cell of an HSR cell sub-block.
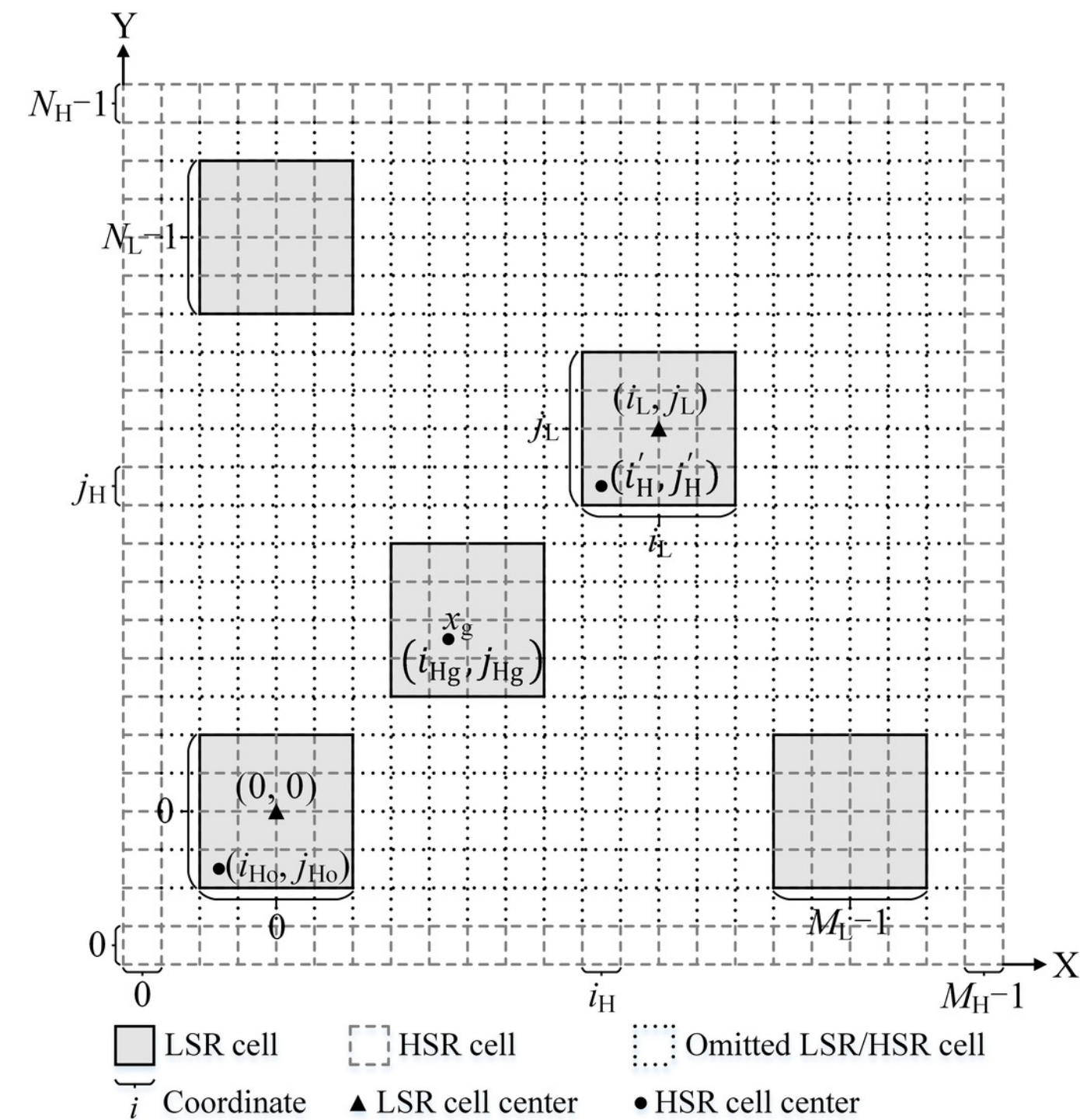
Figure 5

Influence of w_{wc} and w_{sc} on the path.

(A) Influence of w_{wc} on the path. The black region is the obstacle, while the regions with different grayscales from dark to light are the danger region, S_d; the strong constraint region, S_{sc}; and the weak constraint region, S_{wc}. (B) Influence of w_{sc} on the path.
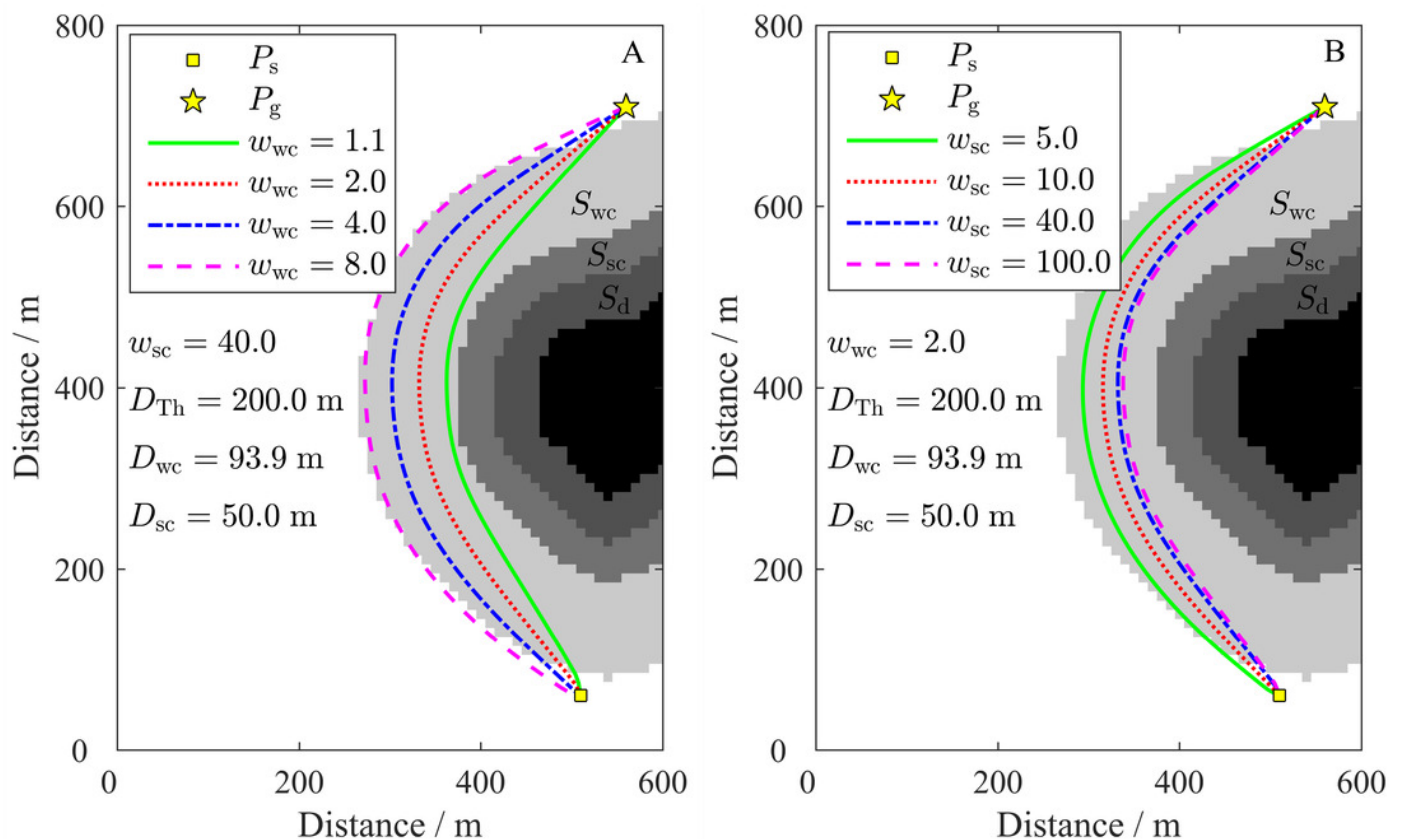
Figure 6

Effective region within the LSR grid map.

The case shown uses $k = 4$, and $P_{ff,i} = (f_{floor}(x_i), f_{floor}(y_i))$, $P_{cf,i} = (f_{ceil}(x_i), f_{floor}(y_i))$, $P_{cc,i} = (f_{ceil}(x_i), f_{ceil}(y_i))$, and $P_{fc,i} = (f_{floor}(x_i), f_{ceil}(y_i))$ are the four neighbor cells of P_i. $P_{ff,i}$ is the "path-passed" cell with respect to P_i in this case.
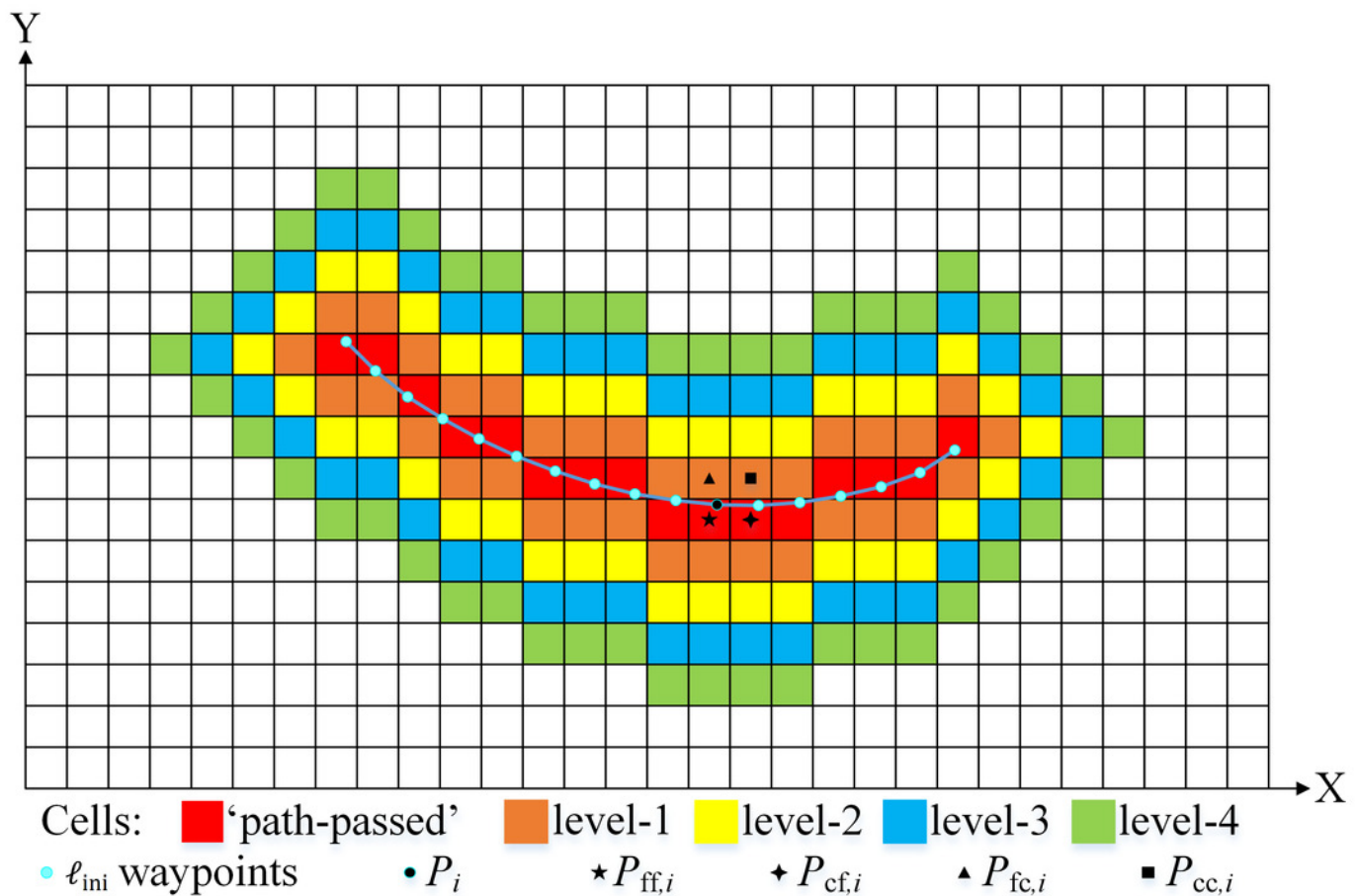
Figure 7

Distances between the corresponding waypoints of l_{ref} and l_k ($k = 3,...,6$).

l_{ref} is the path obtained by applying the IDC-FM^2 method based on the single HSR grid map directly, while l_k is the path obtained by applying the proposed algorithm based on two-level SR grid maps.
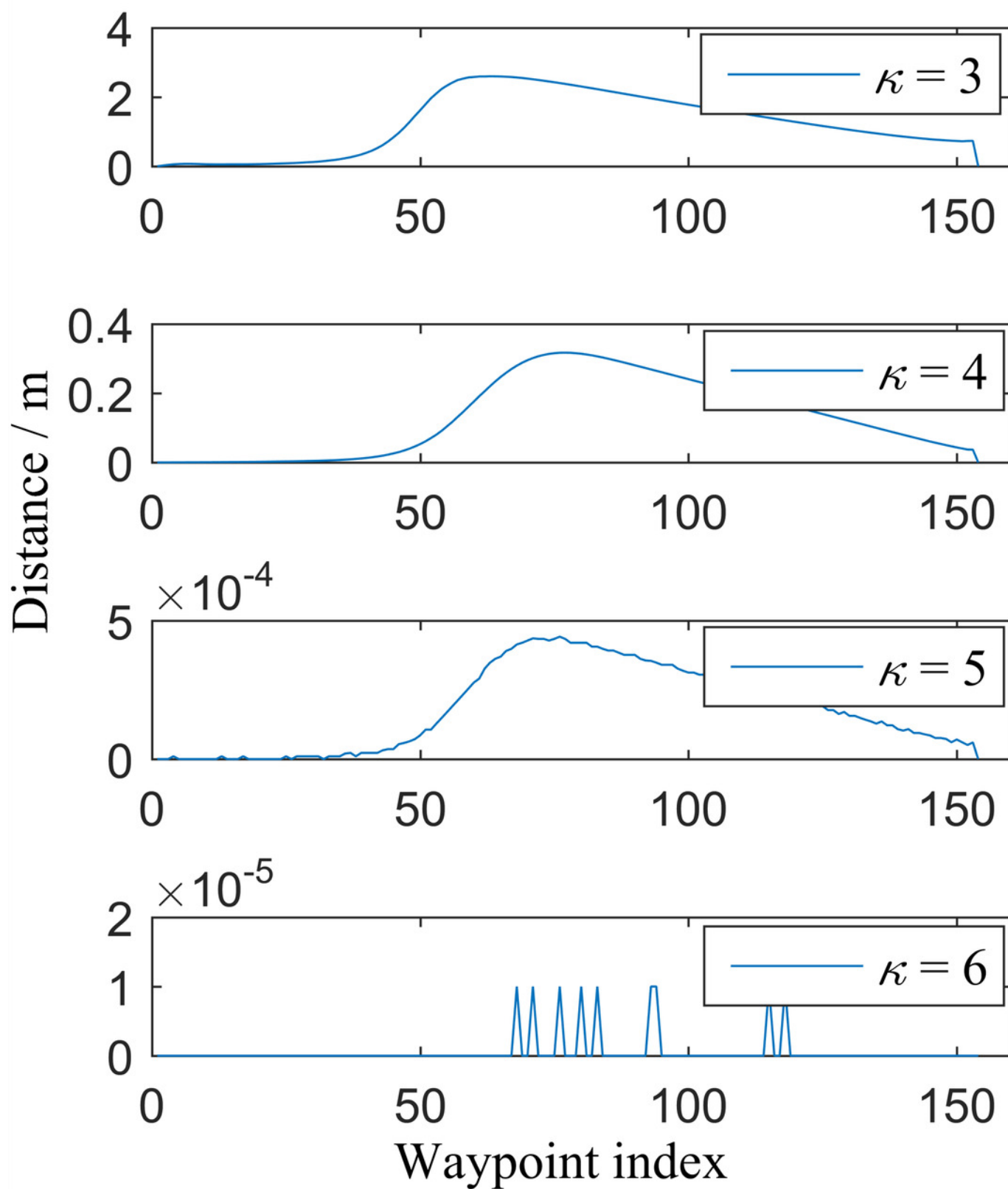
Figure 8

Algorithm flow of the basic algorithm applying the IDC-FM^2 method on a single grid map.

This flow is expressed in the form of main data stream and used methods.
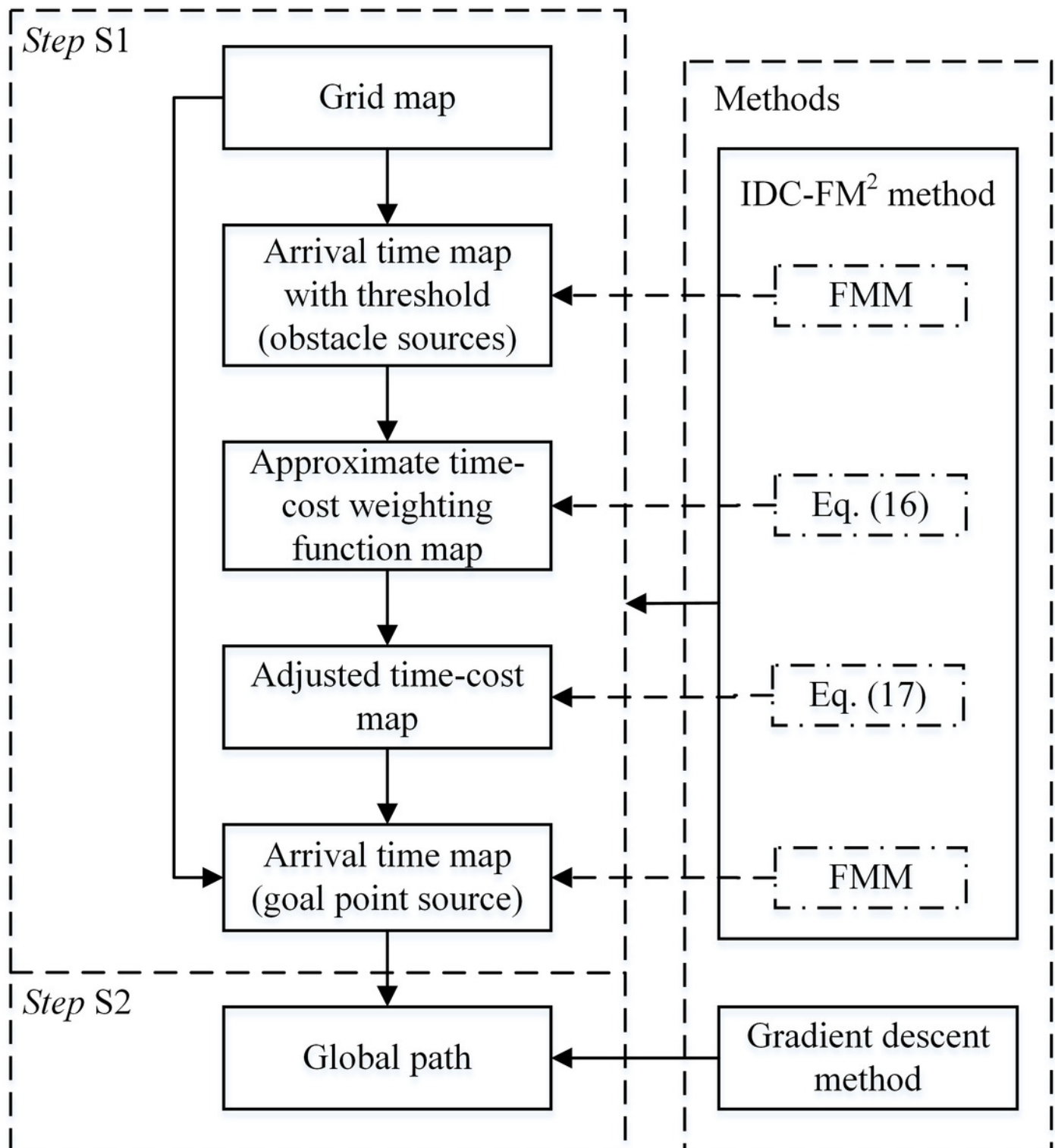
Figure 9

Algorithm flow of the proposed algorithm performed on two-level SR grid maps.

This flow is expressed in the form of main data stream and used methods.
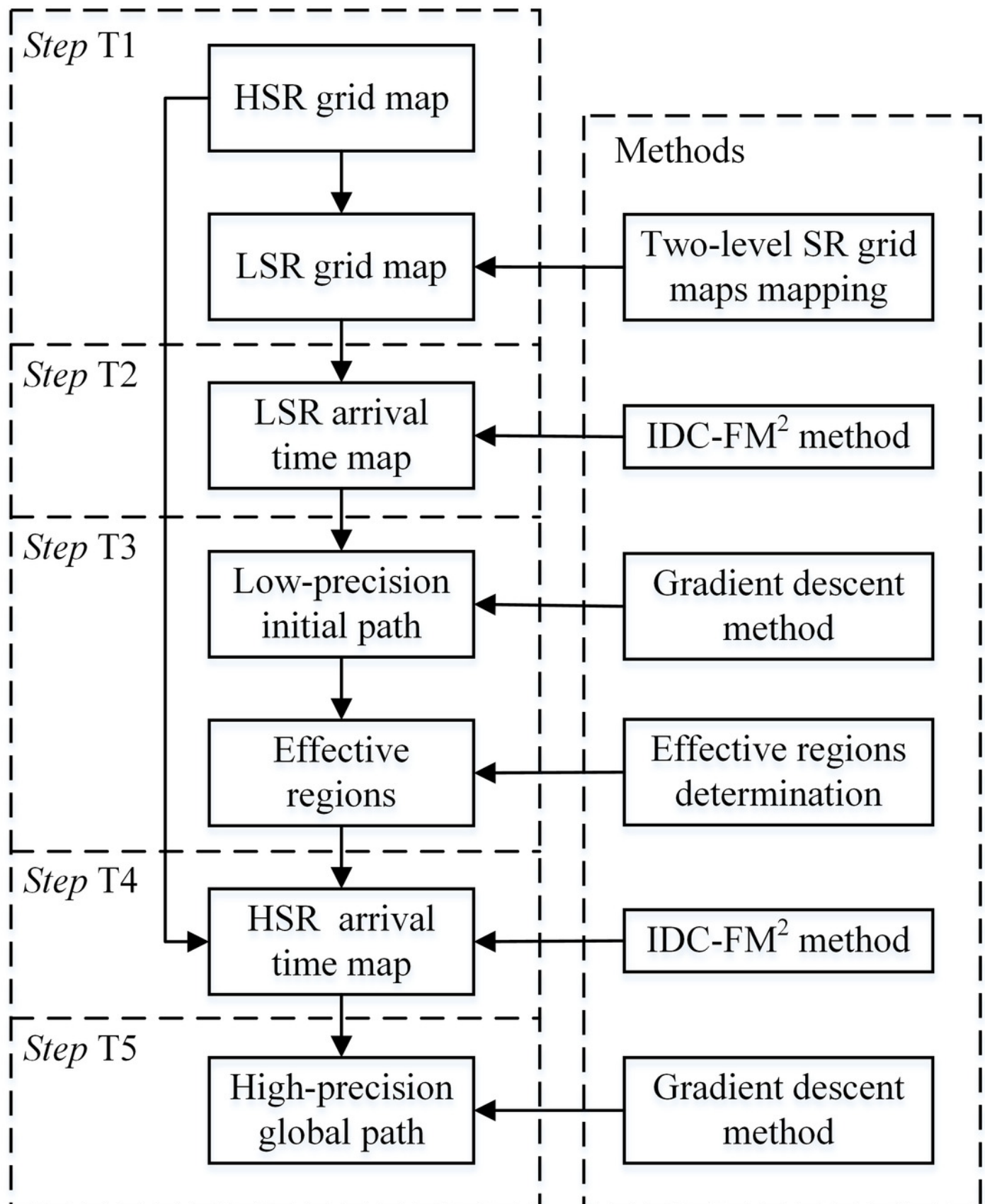
Figure 10

Google satellite images of the simulation environments.

(A) Surrounding areas of Zhucha Island in Qingdao, Shandong Province, China. Map data @ 2020 Google. (B) Surrounding areas of Changhai County in Liaoning Province, China. Map data @ 2020 Google.
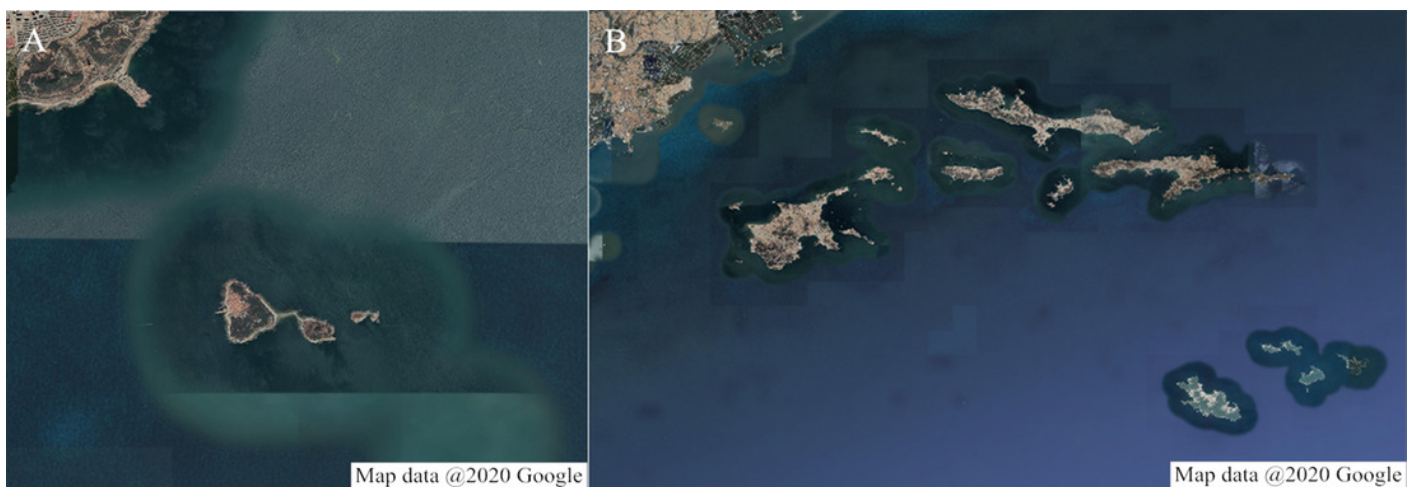
Figure 11

Binary grid maps of the simulation environments.

(A) Binary grid map of the environment shown in *Fig. 10A*. The black cells indicate obstacle areas, while the white cells indicate the navigable areas for USVs. (B) Binary grid map of the environment shown in *Fig. 10B*.
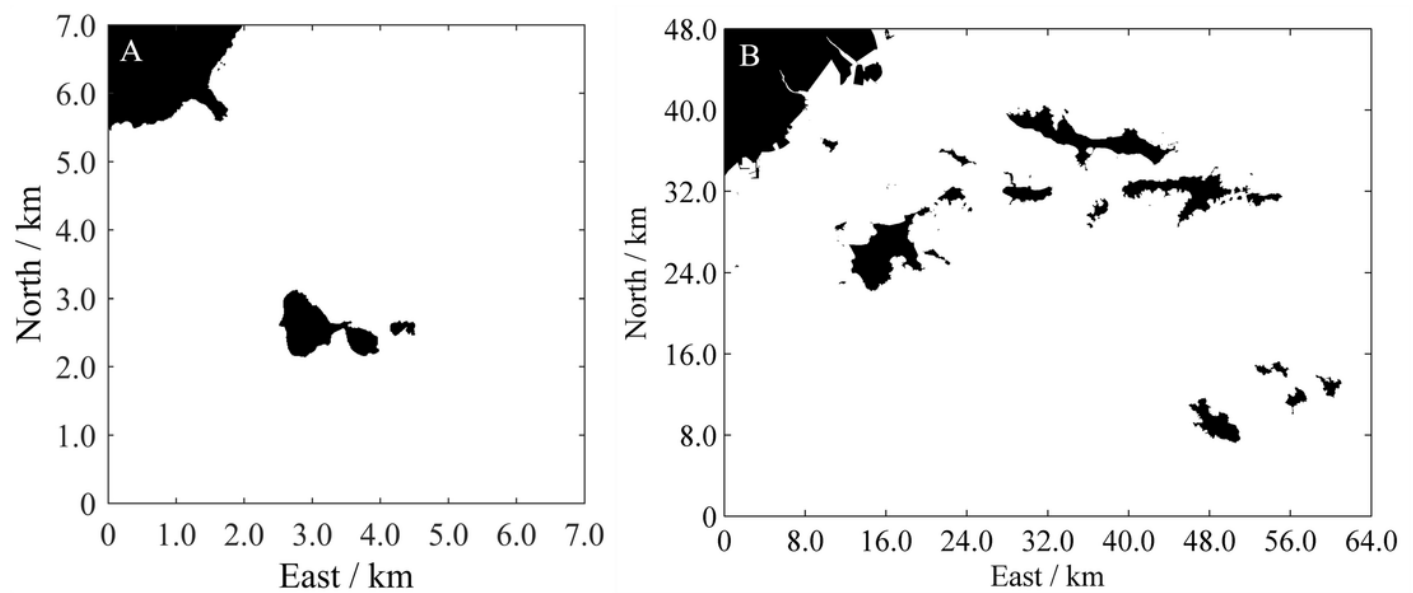
Figure 12

Path based on the basic FMM (l_{FMM}) and paths based on the proposed algorithm with different parameter settings of w_x (l_1 to l_4).

The indicated distances are the estimated quasi-closest distances between the islands and path segments.
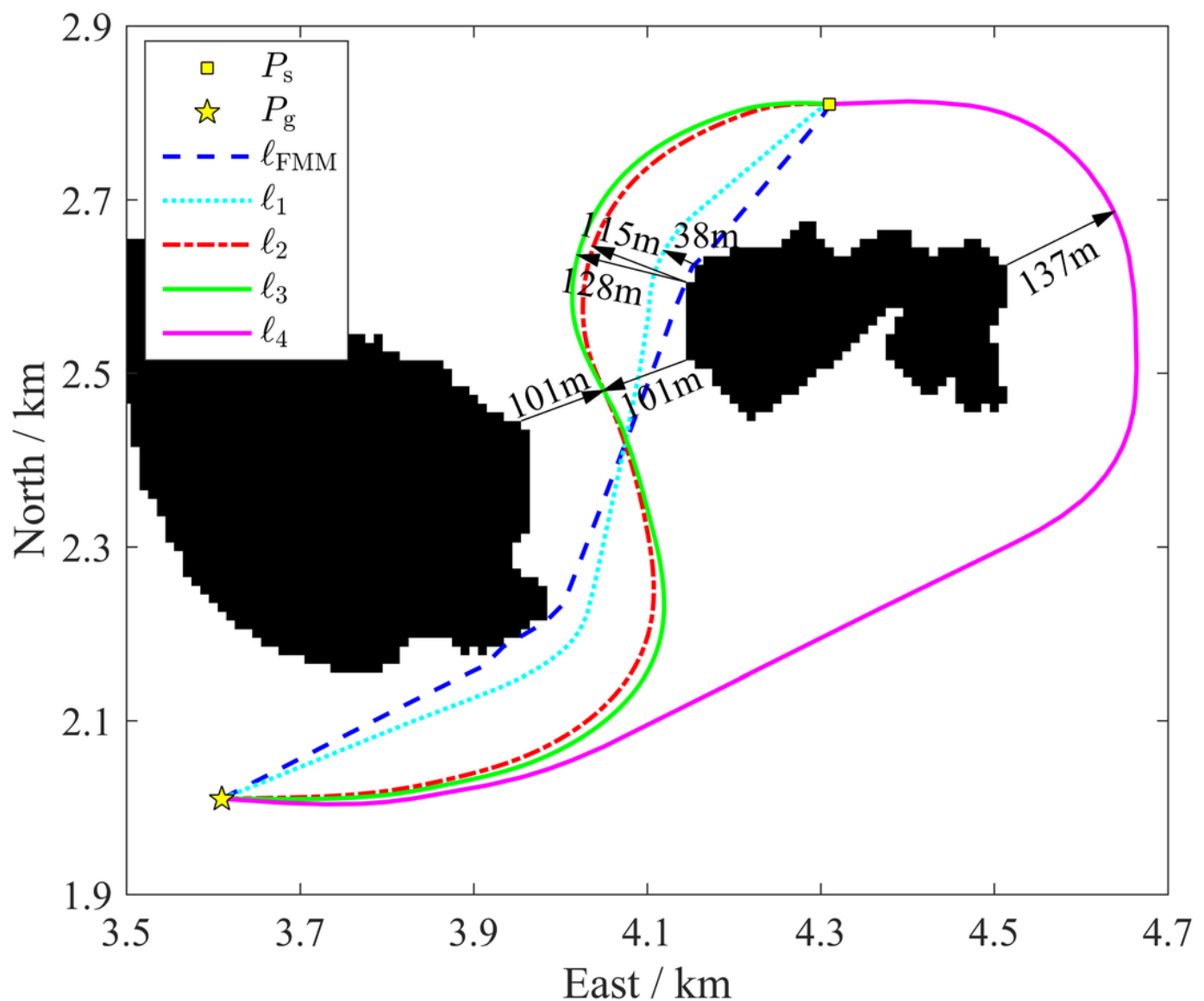
Figure 13

Paths planned by a classical approach applying the FMM within the map with inflated obstacles and the IDC-FM² method.

When using the classical approach, the obstacle is inflated by 94.0 m. The black cells indicate the raw obstacle, the white cells indicate the free areas, while the cells with different gray scales indicate the inflated obstacles.
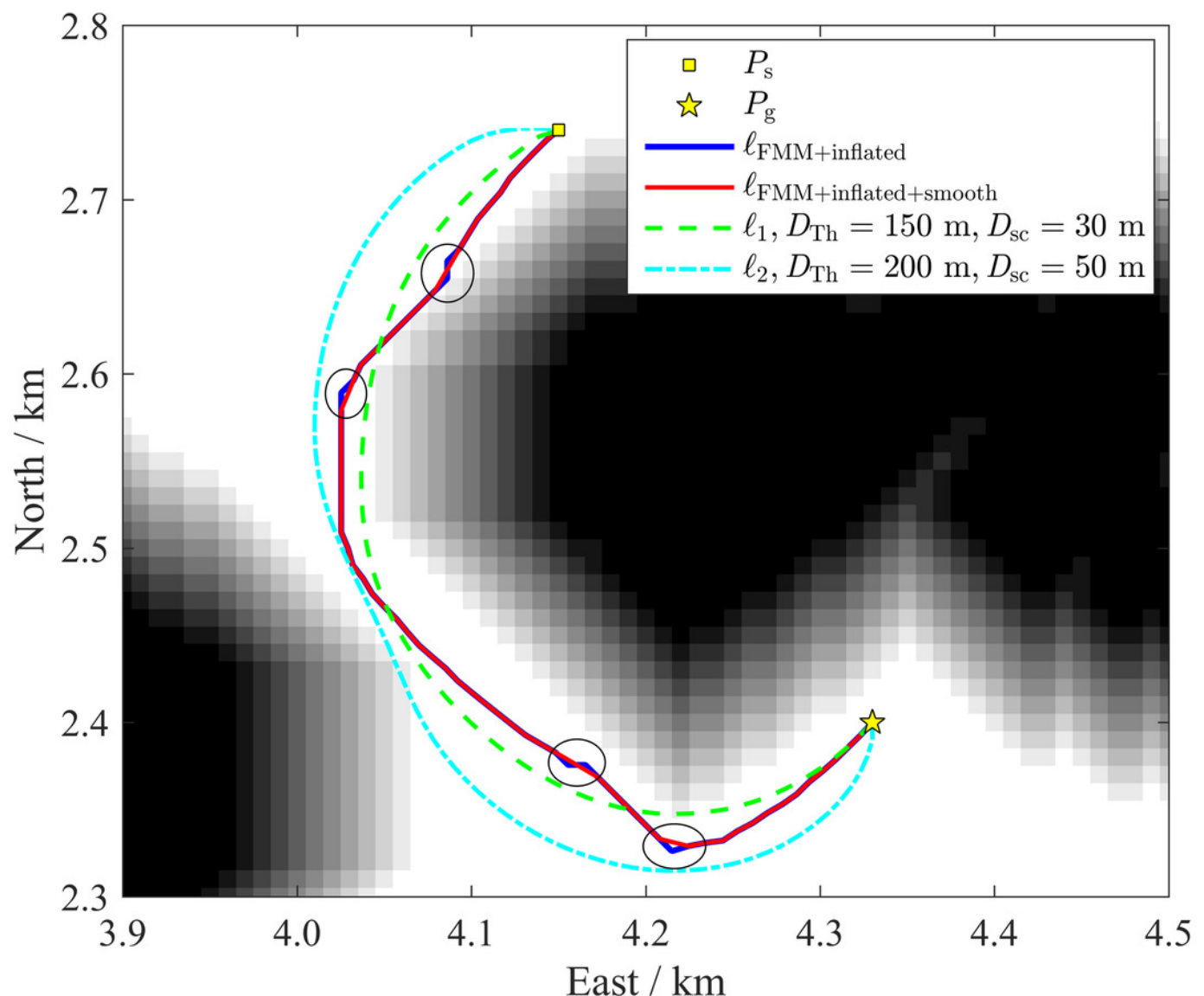
Figure 14

Five typical paths in the surrounding area of Changhai County.

Several portions of these paths marked by rectangles are amplified to show more details in Figure 15.
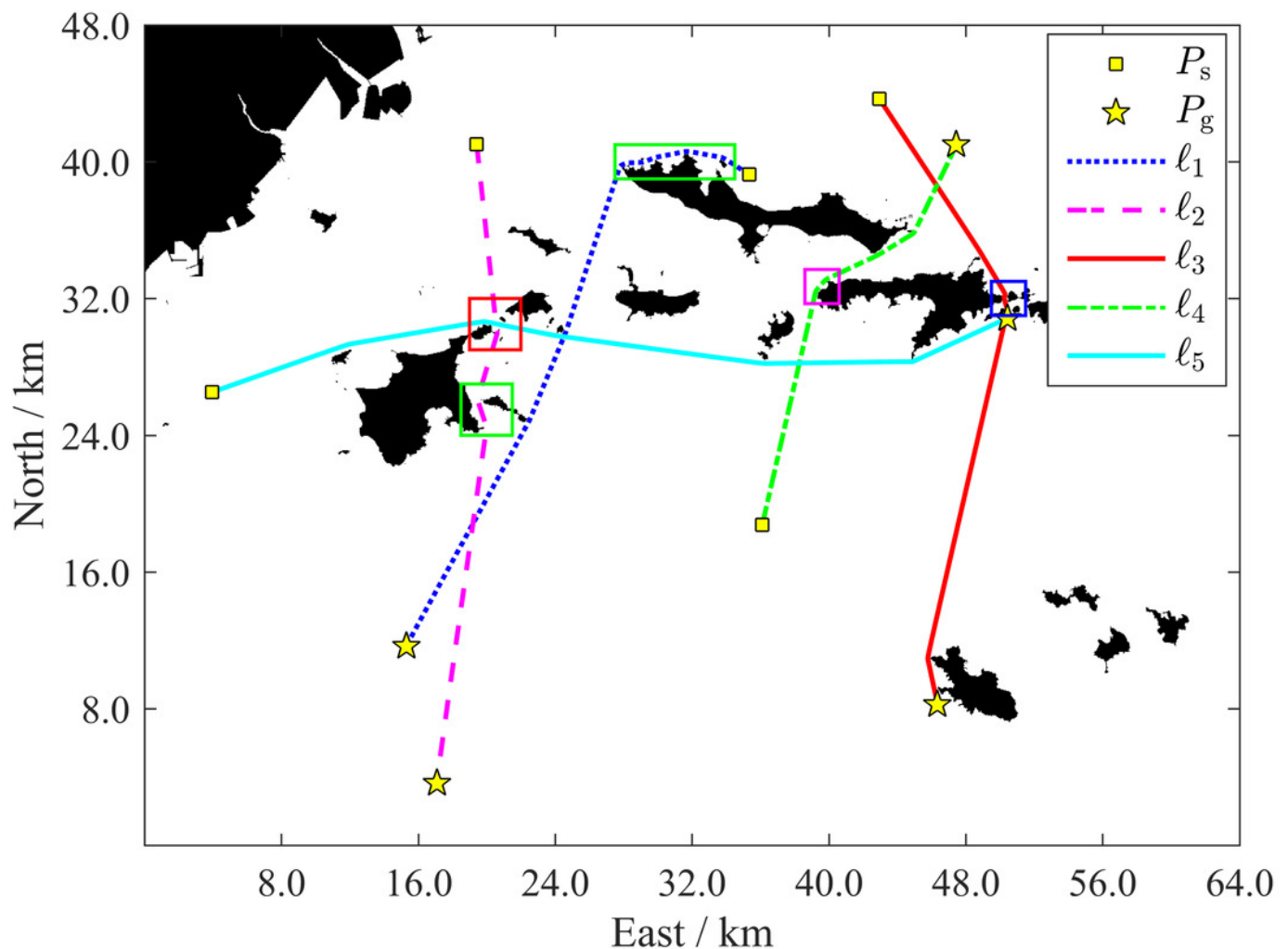
Figure 15

Enlarged views of path portions.

(A)–(E) show the portions that are indicated by rectangles in *Fig. 14*. Detailed information can be acquired from (A) to (E). For example, all paths maintain a relatively safe distance from each island, the paths are smooth even when they turn around islands, and paths across narrow channels can be planned along the quasi-midline of the channels.