

3D-

svetlana_curls@mail.ru

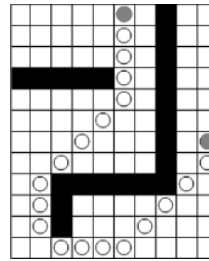
1 4 ()

3D-

1.

() ó

. 1:



2.

4

* [6].

2D-

1 (.1).

3D-

[5]:

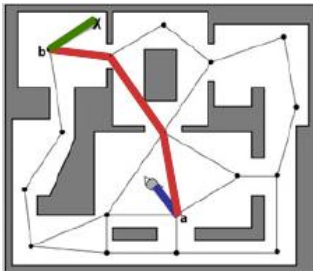
1.

2.

3.

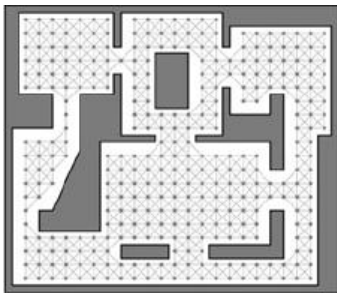
4.

1



. 2:

[3] 3D-



. 3:

navigation mesh [4]
3D-

(. 4).



. 4: navigation mesh.

3.

3D-
 $A_i = \{A_i, i = 1..n\}$ (1), $A_i = \{V, R\}$ (2) ó
 $R = \{R_j, j = 1..m\}$ (3), $R_j = \{V_{j1} \in V, V_{j2} \in V, T, T_D, j_1 \tilde{N} j_2\}$ (4), $T_A \in \{0,1\}$ ó

, T_D

3D-

[8].

S,

P, P_1

1.

$\acute{o} = \{A_i, p_i \tilde{N} p_i', q_i \tilde{N} q_i'\}$ (5), p_i, q_i
 $p_i', q_i' \acute{o} A_i$

2.

3.

4.

5.

() P

bounding box

[1].

$P = \{P_i, P_i \in S, j, k, P_i \in R_j, R_j \in A_k\}$ (6).

6.

<

P

P_1

P' :

$P' = P \cup P_1$ (7).

7.

P_s .

$P_s = \{P_i, D(P_i, P) < D(P_{i+1}, P) \quad i \in 1..N_p - 1\}$ (8),

$D(A, B) \acute{o} P \acute{o} B, N_p$

8.

P_{sk}

$P_{sk} = \{P_i, P_i \in P_s, P_i \in A_k, P_{i+1} \in A_k \quad i \in 1..N_p - 1\}$ (9).

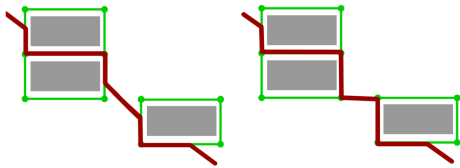
9.

• 1 δ

• 2 δ

T.

10.



.5: ()
()
6

(.5)

2, .2.

1:

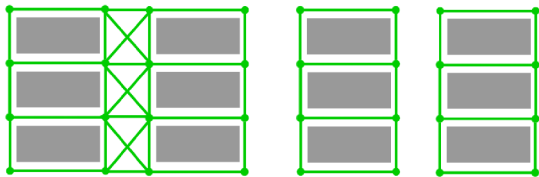
1. . .

(.6).

2.

3.

bounding box



.6: ()

()

4.

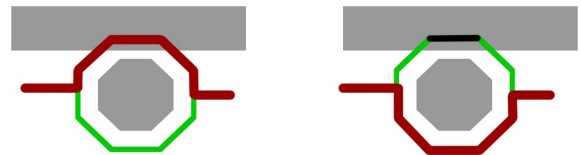
()

4.1

4.2

T_A

(.7).



.7:

()

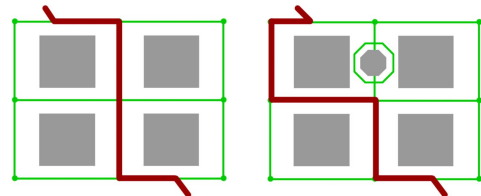
()

4.3

T_D δ

2 :

(.8).

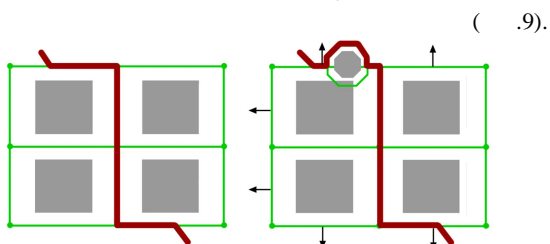


.8:

()

()

• .10



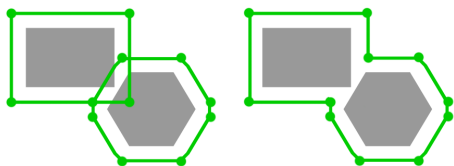
.9: () ()

4.2 4.3

(.2).

4.4

- 1.
- 2.
- 3.



.10: () ()

5.

.4.1,

6.

O(n)

1.		$O(N)$, $N \text{ ó}$
2.		$O(\sum_{i=1}^k \text{len}(V_i))$, $k \text{ ó}$, $\text{len} \text{ ó}$
3.		$O(k*N)$

.1:

(.1),

3 4

.2.

3D-

3D-

« » [7].

7. REFERENCES

[1] A. Botea, M. Muller, and J. Schaeffer, "Near Optimal Hierarchical Path-finding," *Journal of Game Development*, vol. 1, issue 1, 2004.

[2] Xiao Cui, Hao Shi, "A*-based Pathfinding in Modern Computer Games," *IJCSNS International Journal of Computer Science and Network Security*, vol.11, no.1, January 2011.

[3] Mika M., Charla C. Simple, "Cheap Pathfinding," *AI Game Programming Wisdom*, 2002.

[4] John C. O'Neill, "Efficient Navigation Mesh Implementation," *Journal of Game Development*, vol. 1, no. 1, pp. 71-90, 2004.

[5] Paul Tozour, "Fixing Pathfinding Once and For All," 2008. <http://www.ai-blog.net/archives/000152.html>.

[6] Peter Yap, "Grid-Based Path-Finding," *AI '02 Proceedings of the 15th Conference of the Canadian Society for Computational Studies of Intelligence on Advances in Artificial Intelligence*.

[7] <http://vacademia.com>.

[8] <http://www.youtube.com/watch?v=NMunrmtG00o>.