

수심을 고려한 천해역에서의 자동요소 생성법

김남형* · 양정필** · 박상길***

Automatic Mesh Generation Method in Shallow Water Area considering Water Depth

N. H. Kim · J. P. Yang · S. G. Park

Key Words : 자동요소생성(Automatic Mesh Generation), 수정테라우니법(Modified Delauny Method), 천해역(Shallow Water Area), 유한요소망(Finite Element Mesh), 수심 보간법(Depth Interpolation)

Abstract

This paper presents an automatic mesh generation considering water depth, which is based on the depth interpolation. The key feature of this method is that the position of a mesh on any depth in the shallow water area can be generated. The Examples are carried out, and the results are shown to be good. This method is shown to be a useful and powerful tool for the flow calculation for the seabed topography.

1. 서 론

천해역에서 지형변화에 따른 흐름의 해석을 높은 정도로 수행하기 위해서는 적절한 요소의 배치에 의한 모델화가 요구된다. 수치해석에 있어서 가장 많은 노력이 요구되는 공정은 해석하려는 유동

영역(flow field)의 모델화이며, 복잡한 형상의 유동 영역에 관해서 자동적으로 모델화를 수행하는 것은 쉽지가 않다. 따라서 이들 흐름의 해석을 신속하고 또한 높은 정도로 수행하기 위해서는 모델화를 자동적으로 수행하는 방법을 구축하는 것이 천해역에서의 수치해석상 중요하다. 이것에 관해서는

* 정회원, 제주대학교 해양과학대학 해양토목공학과 부교수

** 정회원, 부산대학교 대학원 석사과정 졸업

*** 정회원, 부산대학교 공과대학 토목공학과 교수

많은 연구가 수행되어 왔다(Kono, 1980; Watson, 1981; Sloan, 1986; Taniguchi, 1988; 김·양, 1998).

본 논문은 종래의 자동요소 생성법에 수심을 고려한 것이다. 이에 대한 수치적인 접근은 다음 세 가지로 요약할 수 있다. 첫째는 정선, 섬, 해안·항만구조물 등을 포함하는 내부 및 외부경계에서의 자동생성이고, 두 번째는 수정데라우니법에 의해 새로 생성된 절점에 대해 각 절점에서의 수심 입력 데이터를 유효요소법의 형상함수를 사용하여 수심 값을 계산하고, 셋째로 해석영역에 대해서 복잡한 경계를 가진 영역 전체를 사용자가 임의로 자동요소를 생성하는 것이다. 이것은 입력절점의 데이터를 입력하는 번잡함을 제거해 준다.

본 논문에서는 수면영역의 복잡성과 수심을 고려한 천해역의 계산모델로 광안리해수욕장에서 해운대, 송정, 일광 등을 거쳐 고리까지를 포함하는 동부산권을 모델로 하여 자동요소 생성법을 적용한다. 또 요소로는 임의 형상에 가장 적합성이 좋은 비정규 요소를 이용한다.

2. 자동요소생성

2.1 수정데라우니 삼각분할

수정데라우니 삼각분할은 데라우니 삼각분할에 경계의 개념을 도입하여 국소적인 최소각 최대원리에 근거한다. 이 방법은 절점에 의해 정삼각형에 가까운 형태로 삼각형을 분할하는 것으로, 경계의 생성과 내부의 절점에 의한 삼각분할로 구분할 수 있다(Sloan, S. W., 1986, Sloan, S. W. and G. T. Houlsby, 1984).

그리고 영역내부에서 삼각분할에 사용된 절점은 입력 자료에 의해 설정된 절점(고정점)과 수정데라우니 삼각분할을 수행하는 과정에서 생성된 절점(가동점)이 있으며, 삼각분할 후 가동점에 대해서는 라플라시안법에 의해 영역내부로 절점을 이동시켜 삼각형의 기하학적 형태를 수정하지만 고정

점에 대해서는 절점을 이동시키지 않는다.

수정 데라우니 삼각분할은 주어진 절점에 의해 삼각분할을 수행하기 때문에 엄밀한 경계를 생성하기 위해서는, 각각의 절점군에 대해서 영역을 제한할 필요가 있다. 따라서, 외부 경계상에 위치한 절점군에 대해서는 시계방향으로, 내부경계상에 위치한 절점군에 대해서는 반시계 방향으로 절점을 순차적으로 인식한다. 또 영역내부의 절점에 대해서는 좌표위치로 절점을 인식을 한다. Fig. 1은 내부경계의 생성과정을 나타낸 것으로, 그림 속의 굵은 선은 내부경계를 나타낸다. Fig. 1(a)와 같이 반시계 방향으로 입력된 절점에 대해서 삼각분할을 수행한다(Fig. 1(b)). 그리고 나서, 경계면과 교차하는 요소를 제거하기 위해 생성된 각각의 삼각형을 절점번호 순서로 읽고(Fig. 1(c)), 시계방향으로 읽혀지는 외부영역의 요소를 제거한다(Fig. 1(d)).

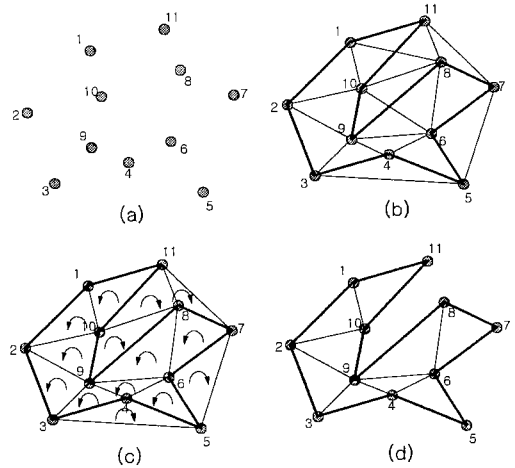


Fig. 1 Triangulation of nodes on a boundary

수심을 고려한 자동요소분할에 관한 수정데라우니법의 알고리즘을 나타내면 Fig. 2와 같이 된다.

새로운 절점을 생성하는 방법은 여러 가지가 있겠으나, 본 연구에서는 평편율법을 이용하여 새로운 절점을 설정한다. 알고리즘의 제3단계(making of rough mesh)까지 생성된 모든 삼각형 요소의

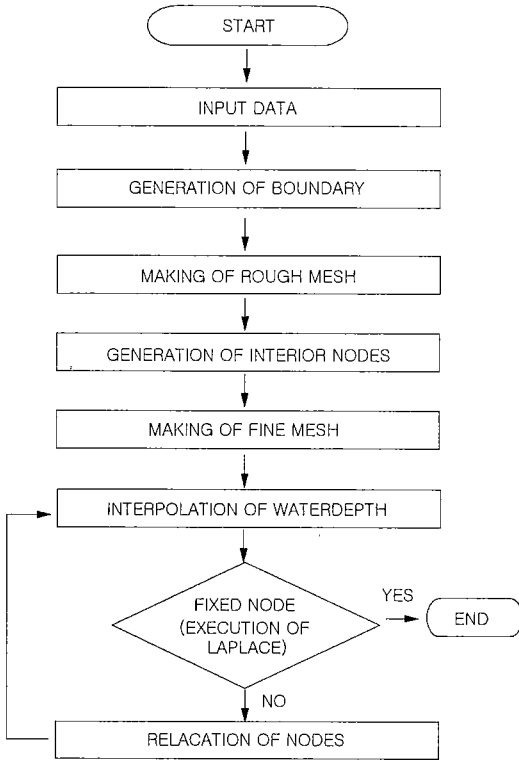


Fig. 2 Flow chart

정각을 계산하여 정각이 큰 삼각형을 찾아내어, 그 삼각형 요소의 정각의 대변에 2등분한 점을 설정하고, 그 점을 이용해서 수정테라우니 삼각분할을 수행한다(김·양, 1997). 그러나 평편율법에 의해 얻어진 삼각형의 형태는 일그러져 있는 일이 많기 때문에 이들 요소형태의 수정이 필요하다. 본 연구에서는 삼각분할의 형태수정을 수행하는 한 가지 방

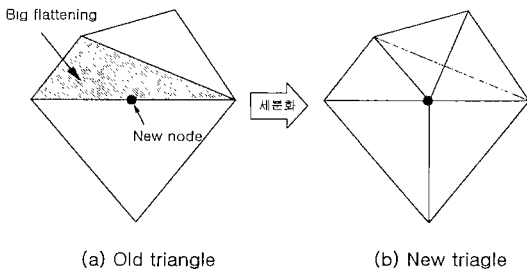


Fig. 3 Flattening method

법으로 라플라시안법을 사용하여 절점을 이동시킨다(Westwood and Holz, 1986).

2.2 수심 보간법

평편율법에 의해 새로 생성된 각 절점에서의 수심 값은 유한요소망(finite element mesh)에 포함된 삼각형의 각 절점 i, j, k 에서의 수심값으로 텐서가 아니며 선형적으로 보간되어 수심 값이 얻어진다(Kashiyama and Kawahara, 1985). 삼각형 요소 X, Y 에서의 수심 h 는 다음과 같이 가정된다.

$$h = \alpha_1 + \alpha_2x + \alpha_3y \quad (1)$$

h_i, h_j, h_k 는 다음과 같다.

$$h_i = \alpha_1 + \alpha_2x_i + \alpha_3y_i$$

$$h_j = \alpha_1 + \alpha_2x_j + \alpha_3y_j \quad (2)$$

$$h_k = \alpha_1 + \alpha_2x_k + \alpha_3y_k$$

여기서,

$$\alpha_1 = \frac{1}{2A} [(x_jy_k - x_ky_j) \Phi_i +$$

$$(x_ky_i - x_iy_k) \Phi_j + (x_iy_j - x_jy_i) \Phi_k]$$

$$\alpha_2 = \frac{1}{2A} [(y_j - y_k) \Phi_i + (y_k - y_i) \Phi_j +$$

$$(y_i - y_j) \Phi_k]$$

$$\alpha_3 = \frac{1}{2A} [(x_k - x_i) \Phi_i + (x_i - x_k) \Phi_j +$$

$$(x_j - x_i) \Phi_k] \quad (3)$$

이며, Φ_i, Φ_j, Φ_k 는 요소의 각 절점에서의 포텐셜 값에 일치하도록 정해진 보간함수이다.

단,

$$2A = \begin{vmatrix} 1 & x_i & y_i \\ 1 & x_j & y_j \\ 1 & x_k & y_k \end{vmatrix}$$

이고, A 는 삼각형의 면적을 나타낸다.

그리고, (3)식의 a_1, a_2, a_3 값을 (2)식에 대입하여 (1)식과 같은 형태로 정리하면 다음과 같이 된다.

$$h = N_i h_i + N_j h_j + N_k h_k \quad (4)$$

여기서, h 는 새로 설정된 좌표에서의 수심이고, h_i, h_j, h_k 는 절점 i, j, k 에서의 수심이다. 그리고, N_i, N_j, N_k 는

$$\begin{aligned} N_i &= \frac{1}{2A} [a_i + b_i x + c_i y] \\ N_j &= \frac{1}{2A} [a_j + b_j x + c_j y] \\ N_k &= \frac{1}{2A} [a_k + b_k x + c_k y] \end{aligned} \quad (5)$$

여기서,

$$\begin{aligned} a_i &= x_j y_k - x_k y_j, & b_i &= y_j - y_k, & c_i &= x_k - x_j \\ a_j &= x_k y_i - x_i y_k, & b_j &= y_k - y_i, & c_j &= x_i - x_k \\ a_k &= x_i y_j - x_j y_i, & b_k &= y_i - y_j, & c_k &= x_j - x_i \end{aligned}$$

이다.

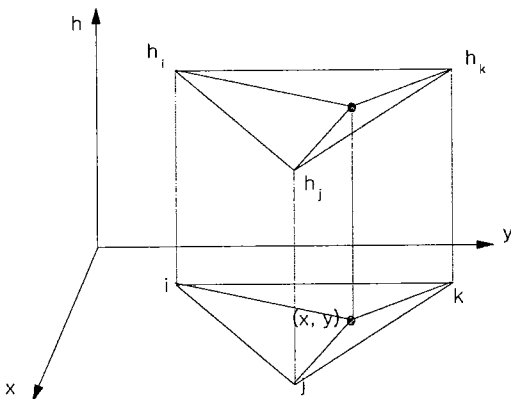


Fig. 4 Depth interpolation

그리고 (4)식을 x, y 에 대해서 미분하면,

$$\frac{\partial h}{\partial x} = \frac{\partial N_i}{\partial x} \phi_i + \frac{\partial N_j}{\partial x} \phi_j + \frac{\partial N_k}{\partial x} \phi_k \quad (6)$$

$$\frac{\partial h}{\partial y} = \frac{\partial N_i}{\partial y} \phi_i + \frac{\partial N_j}{\partial y} \phi_j + \frac{\partial N_k}{\partial y} \phi_k$$

이다. 그러므로, 다음식을 얻는다.

$$\frac{\partial h}{\partial x} = \frac{1}{2A} [b_i h_i + b_j h_j + b_k h_k] = const. \quad (7)$$

$$\frac{\partial h}{\partial y} = \frac{1}{2A} [c_i h_i + c_j h_j + c_k h_k] = const.$$

(7)식에 알 수 있듯이, 편미분을 취한 값이 일정하기 때문에 요소 내에서의 기울기가 일정하며, 요소 자동분할 수행 시에 수심 h 가 급격히 변화하는 영역에서는 요소의 크기를 작게 취해야 정확한 수심 값을 얻을 수가 있다(Larry, 1984).

3. 수치계산의 결과와 고찰

3.1 적용예

Fig. 5는 수심을 고려한 수정테라우니 삼각분할을 적용할 간단한 계산 모델을 나타내며, 이 모델은 가로 15m, 세로 2m, 해저경사 1:7.5인 가상적인 수로다.

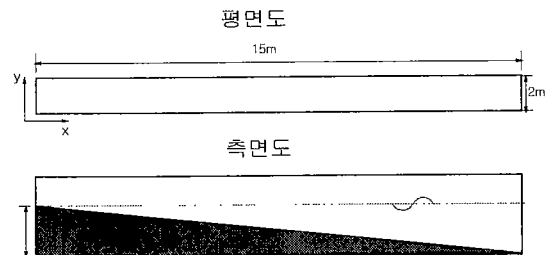


Fig. 5 Test model



(a)

(Boundary node: 32, Input nodes:0, Total nodes:32, Elements:30)



(b)

(Boundary node: 32, Input nodes:17, Total nodes:49, Elements:62)

Fig. 6 Finite element mesh generated by test model

Fig. 6은 수심을 고려한 수정테라우니 삼각분할을 수행하였을 때 입력 절점만에 의한 요소분할의 결과와 절점 17개를 생성했을 때의 요소분할의 결과를 비교하고 있다. Fig. 5에서 나타난 초기조건으로 하여 입력절점만을 이용하여 자동요소 생성을 수행한 결과, 경계가 정확하게 생성되고 있으며, 이에 대한 결과를 Fig. 6(a)에서 나타내고 있다. 여기서 입력절점의 수는 32개, 요소수는 30개이다. Fig. 6(b)은 사용자에게 의해 입력절점 17개를 생성하여 총 절점수(Total node)가 49개일 경우 요소분할을 수행한 결과를 나타내고 있다. 그림에서 알 수가 있듯이 새로운 절점을 추가시킴으로서 정삼각형에 가까운 조밀하고 모양 좋은 삼각형 요소를 얻을 수 있었다. 이 경우, 삼각형 요소 62개가 생성되었다. 하지만, Fig. 6(b)에서의 분할은 어디까지나 삼각형 요소형태를 보기 좋게 하기 위해 요소분할을 수행한 결과이고, 실제 계산에서는 이보다 더 복잡한 요소분할이 요구가 된다.

Table 1은 Fig. 10(b)의 수치실험 결과 생성된 절점의 좌표와 수심 값에 대한 결과를 나타내고 있다.

Fig. 7은 Table 1의 수치계산 결과 계산된 x, y 의 절점좌표와 수심 값을 나타내며, Fig. 5에서 가정한 하상경사(bottom slope)가 1:7.5일 때와 비교

Table 1 Numerical data calculated by test model

NO	x-coordinate (m)	y-coordinate (m)	depth (m)
1	10.50	1.00	-2.10
2	8.50	1.00	-1.70
3	0.65	1.00	-0.13
4	5.50	1.00	-1.10
5	7.50	1.00	-1.50
6	9.50	1.00	-1.90
7	6.50	1.00	-1.30
8	14.35	1.00	-2.87
9	0.00	1.00	0.00
10	3.50	1.00	-0.70
11	12.49	1.00	-2.50
12	15.00	1.00	-3.00
13	1.54	1.00	-0.31
14	11.50	1.00	-2.30
15	2.51	1.00	-0.50
16	4.50	1.00	-0.90
17	13.46	1.00	-2.69

하고 있다. Fig. 7로부터 새로 생긴 절점에서의 수심 값은 하상경사의 기울기와 일치하고 있고, 또한 수심값을 정확하게 계산하고 있는 것을 알 수 있

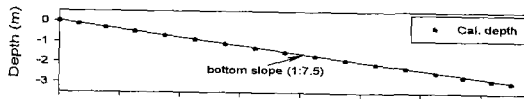


Fig. 7 Comparison between cal. depth and bottom slope

다. 그러므로 본 논문에서 제안한 수심을 고려한 수법은 천해역에서 수치계산을 위한 자동요소생성에 대해 적용할 수 있겠다.

3.2 수치계산

본 연구를 위한 계산 영역은 해석영역이 복잡한 광안리해수욕장에서 해운대 해수욕장, 송정, 일광에서 고리까지를 포함한 동부산권을 모델로 하여 연구의 유용성을 검토하기로 했다. Fig. 8은 해석영역의 수심도를 나타낸 것으로 국립해양조사원에서 발행한 No.142, No.202지도를 사용하였다.

해석영역에 대한 수치실험의 초기조건과 경계면 상에서의 입력 절점의 조건과 그 결과를 Table 2와 Fig. 9 ~ Fig. 12에 나타낸다.

Table 2 Numerical data in the calculating area

Case	Boundary nodes	Interior nodes	Input nodes	Total nodes	Elements
Case A	120	200	200	520	886
Case B	130	200	200	530	902
Case C	140	200	200	540	914

Fig. 12은 Fig. 11의 절점을 이용하여 해석영역에 대한 등수심도로 외부 경계상의 수심이 깊은 부근(offshore)인 수심 80~100m부근에서 등수심선이 일치하지 않는 부분은 있지만, 해안선과 전체적인 등수심도가 대체로 Fig. 8과 일치하고 있다. 여기서, 수심이 깊은 외해부근의 외부경계 쪽에서 실제

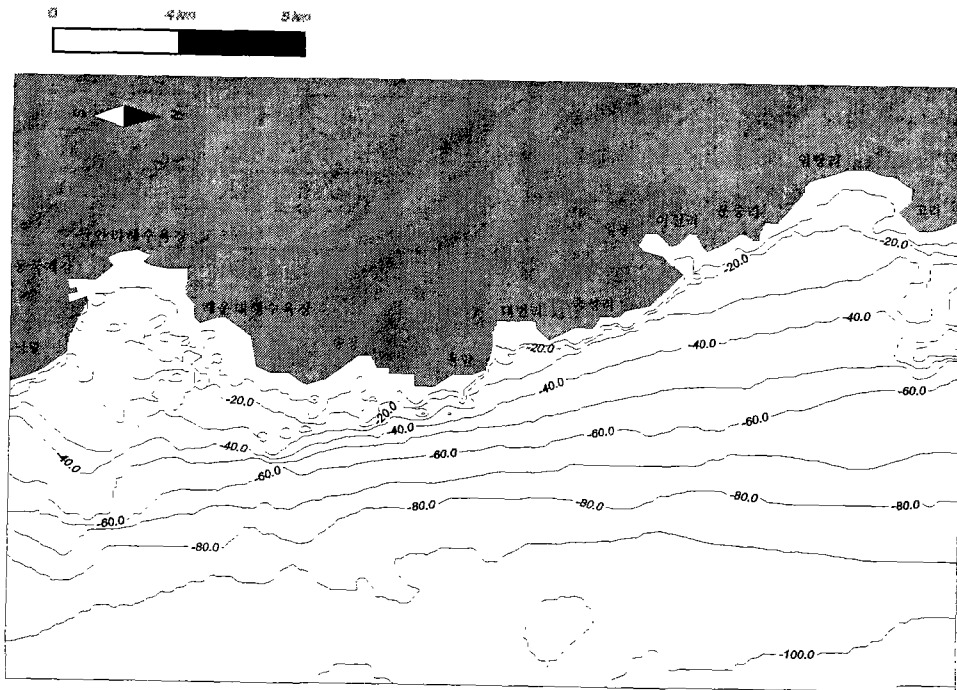


Fig. 8 Isobath map in the Eastern-Pusan

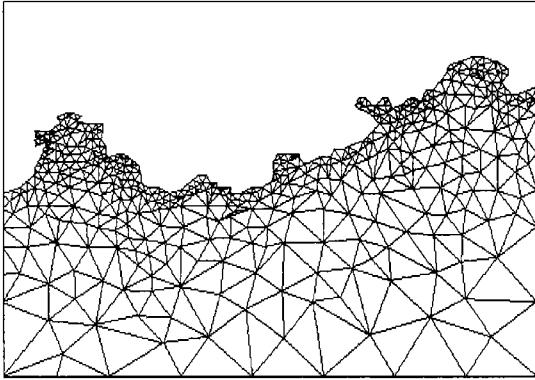


Fig. 9 Finite element mesh generated by the case A
(Interior nodes:200, Boundary nodes:120, Input nodes:200, Elements:886)

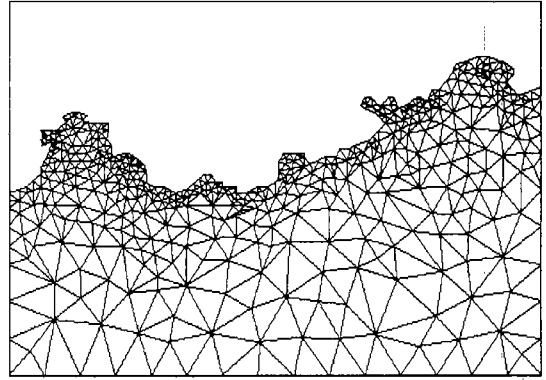


Fig. 11 Finite element mesh generated by the case C
(Interior nodes:200, Boundary nodes:140, Input nodes:200, Elements:914)

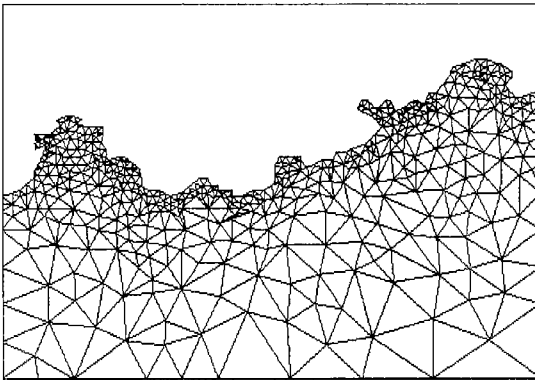


Fig. 10 Finite element mesh generated by the case B
(Interior nodes:200, Boundary nodes:130, Input nodes:200, Elements:902)

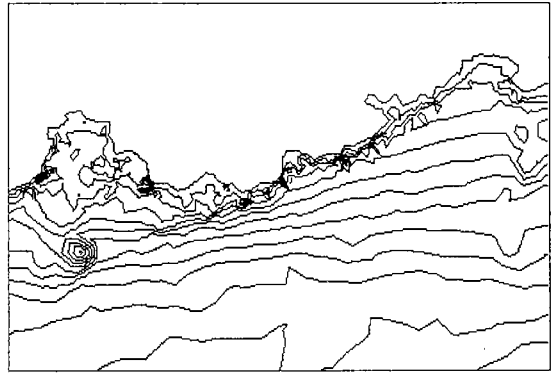


Fig. 12 Contour used to nodes generated in the case C

지형과 차이를 보이고 있는 것은 이 부분에서의 입력절점과 수정테라우니 삼각분할에 의해서 생성된 절점의 수가 작아 정확한 등수심도를 그리지 못하기 때문으로 생각된다. 그러므로 외해쪽에서 등수심도를 정확하게 그려내기 위해서는 보다 많은 절점의 입력이 필요하다.

이들 수심 값을 보다 구체적으로 비교하기 위해

수정 테라우니 삼각분할에 의해 생성된 수심 값과 현지 지형의 수심 값을 임의로 추출하여 비교한 것을 Fig. 13에서 나타낸다.

Fig. 13은 Fig. 11의 수정테라우니 삼각분할의 수치계산 결과로서 얻은 수심 값을 임의로 70개를 추출하여 수심값이 작은 순서로 배열하여 새로 생성된 절점에서의 실제 수심 값과의 관계를 비교하고 있다. Fig. 13에서 알 수 있듯이 수정테라우니 삼각분할에 의해 생성된 수심 값과 현지 지형에서

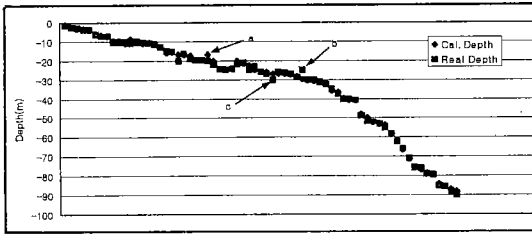


Fig. 13 Comparison between cal. depth and real depth

의 수심 값이 거의 일치하고 있음을 알 수가 있고, a, b, c에서와 같이 수심 값이 차이를 보이고 있는 것은 절점이 생성된 부분이 해저지형의 굴곡이 심하거나 급격한 수심변화가 있는 해구, 해령 등이 있는 부근을 나타낸다. 따라서, 이러한 경우를 피하기 위해서는 입력 데이터의 입력 시에 수심이 급격하게 변화하는 부분에 입력 자료에 의한 내부절점을 설정해 주면 이러한 수심 값에 대한 오차는 해소가 될 수 있을 것으로 사료된다.

Fig. 14은 Case A와 Case B, Case C에 대해서 전체 절점수와 삼각형 요소수와의 상관관계를 나타낸 것이다.

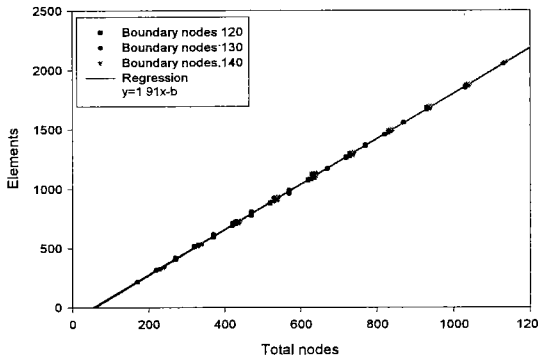


Fig. 14 Relationship between total nodes and element

Fig. 14에서 수정 테라우니 삼각분할을 수행한 모든 Case에 대하여 $y = 1.91x + b$ 의 선형함수로 나타나고 있다. 여기서 알 수 있는 것은 경계상의

입력절점의 수를 120, 130, 140개까지 증가시켜도 삼각형 요소수와의 관계가 적으며, 또 선형함수와 의 관계도 잘 일치하고 있는 것으로 보아, 입력 데이터에 의해 경계상에 설정된 절점은 요소분할 수행시 보다 균형 있고 조밀한 삼각형을 형성하는데 도움이 되는 것으로 나타나고 있다.

4. 결론

본 연구에서는 수심을 고려한 수정 테라우니 삼각분할을 이용한 2차원 유동에 대해 천해역에 있어서 요소자동생성법을 제안했다. 수치실험 결과 해안선 부근과 같이 유동영역이 복잡한 영역에 대해서도 요소분할이 자동적으로 수행되었고, 입력데이터를 제어하는 것에 의해 사용자가 원하는 요소분할의 형상을 제어할 수 있었다. 따라서 천해흐름의 문제에 따라 알맞게 절점을 배치할 수 있어 보다 균형 잡힌 요소모델의 설정이 가능하였고, 수심보 간법에 의해 생성된 수심 값이 정확하게 계산되었으며, 수심 데이터에 대한 입력의 간편화를 통해 이를 이용한 천해 흐름의 유동을 수치해석하는데 걸리는 요소작성의 시간을 줄일 수 있었다.

따라서 종래의 천해역에서의 유동계산에서는 수심이 가장 중요한 파라미터임에도 불구하고 수작업, 또는 수심을 고려하지 않은 요소분할을 수행하여 유동계산에 적용하여 왔지만, 수심을 고려한 요소의 자동생성법을 적용하면 천해역의 유동해석에 있어서 보다 높은 정도의 수치계산을 할 수 있겠다.

참고문헌

- 1) Kono, M., A generalized automatic mesh generation scheme for finite element method, Int. J. Num. Meth. Engng, Vol. 3, pp. 713-731, 1980.

- 2) Watson, D.F., Computing the n-dimensional delaunay tessellation with application to voronoi polytopes, The computer journal, Vol. 24, No. 2, pp. 167-172.
- 3) Sloan, S. W., A fast algorithm for constructing delaunay triangulations in the plane, Adv. Eng. Software, 9(1), pp. 34-55, 1986.
- 4) Taniguchi, T., An interactive automatic mesh generator for the microcomputer, Computers and Structures, Vol. 30, No. 3, pp. 715-722, 1988.
- 5) 김남형 · 양정필, 천해역에서의 유한요소 해석을 위한 자동요소분할법, 제주대학교, 해양연구 논문집 제21권, pp. 33-42, 1997.
- 6) Sloan, S. W. and G. T. Houlsby, An implementation of Watson's algorithm for computing 2-dimensional Delaunay triangulations, Adv. Eng. Software, 6(4), pp. 192-197, 1984.
- 7) Westwood, I. J. and K. P. Holz, Automatic Optimization of Irregular Triangular Meshes for Natural Flow Computation. International conference hydrosoft 86, pp. 423-435, 1986.
- 8) Kashiyama, K. and M., Kawahara, Interpolation method for preparation of input data of water depth in finite element analysis of shallow water flow, Eng. comput. Vol. 2, pp. 266-270, 1985.
- 9) Larry, J., Applied finite element analysis, John wiley and sons, pp. 56-60, 1984.