HOKKAIDO UNIVERSITY

Title	Experimental and Theoretical Analyses for Microwave Backscattering from Lake Ice in Northern Alaska
Author(s)	Wakabayashi, Hiroyuki
Degree Grantor	北海道大学
Degree Name	博士(工学)
Dissertation Number	乙第5015号
Issue Date	1996-06-28
DOI	https://doi.org/10.11501/3113528
Doc URL	https://hdl.handle.net/2115/51354
Type	doctoral thesis
File Information	000000298719.pdf

Hokkaido University Collection of Scholarly and Academic Papers : HUSCAP

Experimental and Theoretical Analyses for Microwave Backscattering from Lake Ice in Northern Alaska

HIROYUKI WAKABAYASHI

Experimental and Theoretical Analyses for Microwave Backscattering from Lake Ice in Northern Alaska

(アラスカ北部湖氷のマイクロ波後方散乱に関する実験的及び理論的解析)

Hiroyuki Wakabayashi

若林　裕之

A Dissertation
Presented to the Graduate Committee
of Hokkaido University
in Candidacy for the Degree of
Doctor of Philosophy

National Space Development Agency of Japan
1996

Abstract

This paper describes the C- and L-band backscattering variations and the development of a numerical model of backscatter from lake ice growing on shallow tundra lakes in northern Alaska. This research was prompted by earlier observations of unusual X– and L–band backscatter variations in side–looking airborne radar (SLAR) data, and the recent availability of a well–calibrated time–series of SAR data and concurrent ground observations and measurements.

The North Slope of Alaska is a large expanse of low–lying tundra with many shallow lakes which comprise more than 40 % of the surface area. During the 1970s, in late winter, SLAR images of these lakes showed interesting variations in backscatter intensity: areas of low backscatter at lake margins were believed to represent ice that was frozen to the lake bed while areas of high backscatter represented floating ice that contained tubular bubbles which acted as forward scatterers.

From September 1991 to April 1993, backscatter intensity variations from shallow tundra lakes near Barrow, NW Alaska, were studied using C- and L-band SAR data from the ESA Remote-Sensing Satellite-1 (ERS-1) and Japanese Earth Resource Satellite-1 (JERS-1). The SAR data were processed at the Alaska SAR Facility (ASF) and Earth Observation Center (EOC), and the backscattering coefficient was derived for a number of lakes. Field measurements in April 1992 confirmed that the highest values were associated with floating ice containing tubular bubbles, while the lowest backscatter values were associated with ice frozen to the lake bed. The ice frozen to the bottom of the lakes also contained tubular bubbles.

Ice core measurements indicated that the lake ice comprised three layers: 1) a surface layer of granular ice with roughly spherical bubbles with radii smaller than a wavelength; 2) a layer of clear bubble-free ice, and; 3) a bottom layer containing tubular bubbles resembling thin cylinders with lengths ranging from 15 mm and 91 mm and a radius much smaller than a wavelength. The number of tubular bubbles per square

meter, i.e. bubble density or ice porosity, was quite variable.

The backscatter model that has been developed comprises the following elements: 1) an ice layer of variable thickness; 2) ice sub–layers with air inclusions of variable density, size and shape, including cylinders of finite length, prolate spheroids and spheres; 3) air-ice boundary is a specular surface, while ice–water and ice–frozen soil boundaries are specular or slightly rough.

The model results confirm that backscatter is a sensitive function of the presence of an ice–water interface, with a roughly 30–times greater backscatter intensity than from an ice–frozen soil interface. The model also shows that backscatter increases as the ice grows. The model has also been tested using bubble data derived from ice cores in April 1992. The modelled backscatter are compared with backscatter derived from ERS–1 SAR images obtained at the same time as the fieldwork. It has been found that the modelled and actual backscatter have high correlation.

CONTENTS

Abstract i

List of Figures v

List of Tables vii

CHAPTER 1. INTRODUCTION 1

CHAPTER 2. LAKE ICE MEASUREMENT AND CHARACTERISTICS 5

2-1. TEST SITE 5

2-2. FIELD MEASURED DATA 7

2-3. ICE CORE MEASUREMENT 8

2-4. ICE CHARACTERIZATION 17

CHAPTER 3. SAR DATA FOR LAKE ICE 19

3-1. SAR DATA 19

3-1.1. C-BAND SAR DATA 20

3-1.2. L-BAND SAR DATA 28

3-2. TEMPORAL CHANGES IN BACKSCATTER 34

3-2.1. TIME CHANGES FOR FIELD OBSERVATION SITES IN C-BAND 34

3-2.2. SPATIAL AND TEMPORAL BACKSCATTER CHANGE AT LAKE SUKOK IN C-BAND 37

3-2.3. TIME CHANGES FOR FIELD OBSERVATION SITES IN L-BAND. 40

3-3. ICE GROWTH, STRUCTURE AND BACKSCATTER VARIATIONS 42

3-3. 1. C-BAND BACKSCATTER 42

3-3. 2. L-BAND BACKSCATTER 44

CHAPTER 4. LAKE ICE BACKSCATTER MODEL 45

4-1. LAKE ICE FEATURES CONSIDERED IN MODELING 45

4-2. AVERAGE DIELECTRIC CONSTANT FOR ICE 47

4-2.1. TWO-PHASE MIXTURE CONTAINING ELLIPSOIDAL INCLUSIONS 48

4-2.2. MIXTURE MODEL FOR LAKE ICE....49
4-3. REFLECTIVITY OF BOUNDARY....50
4-4. POWER LOSS FACTOR OF ICE....52
4-5. BUBBLE SCATTERING....53
4-5.1 TUBULAR BUBBLES WITH LENGTH GREATER THAN WAVE LENGTH....54
4-5.2 TUBULAR BUBBLES WITH LENGTH SMALLER THAN WAVE LENGTH....56
4-5.3 SPHERE BUBBLES....58
4-6. MULTI-LAYER CONSIDERATION....58
4-7. SURFACE SCATTERING FROM ICE-BOTTOM BOUNDARY....64
CHAPTER 5. BACKSCATTER SIMULATION....67
5-1. MODEL COMPONENT....67
5-1.1. AVERAGE DIELECTRIC CONSTANT....67
5-1.2. REFLECTIVITY AT ICE-BOTTOM BOUNDARY....69
5-1.3. BUBBLY ICE LAYER....71
5-1.4. SURFACE ROUGHNESS AT ICE-BOTTOM BOUNDARY....74
5-2. MULTI-LAYER SIMULATION....76
5-2.1. FLOATING AND GROUNDED ICE....76
5-2.2. GROWING ICE....78
5-3. COMPARISON BETWEEN ERS-1 SAR AND MODELED BACKSCATTER....80
5-3.1. SPECULAR ICE-BOTTOM BOUNDARY....80
5-3.2. ROUGH ICE-BOTTOM BOUNDARY....82
CHAPTER 6. CONCLUSIONS....84
ACKNOWLEDGMENTS....87
Reference....88
Appendix....92

List of Figures

Fig. 2-1 Map of the region around Barrow including field observation locations. 6

Fig. 2-2 Diagrams for ice cores illustrating the amount and distribution of ice types..... 13

Fig. 2-3 Sample photographs for ice cores in sub layers 14

Fig. 2-4 The porosity profile for each test site..... 15-16

Fig. 3-1 Time series of C-Band radar images for lakes near Barrow. 22-27

Fig. 3-2 Time series of L-Band radar images for lakes near Barrow. 30-33

Fig. 3-3 C-band backscatter changes as a function of time for a selection of lakes during the period from 29 September 1991 to 20 April 1992..... 36

Fig. 3-4 Profiles of spatial changes in digital number σ^0 values along the same transact across Sukok Lake from 29 September 1991 to 20 April 1992..... 38-39

Fig. 3-5 L-band backscatter changes as a function of time for a selection of lakes. 41

Fig. 4-1 Typical structure of lake ice as observed in Northern Alaska. 47

Fig. 4-2 Configuration of two-phase mixture model. 48

Fig. 4-3 Reflection and transmission at a specular boundary. 51

Fig. 4-4 Geometry of bubbles considered in the lake ice model. 54

Fig. 4-5 Schematic figure of lake ice reflection including direct backscattering and forward scattering from bubbles. 62

Fig. 4-6 Multi-layer ice structure. 63

Fig. 4-7 Backscattering from rough ice-bottom boundary..... 66

Fig. 5-1 Average dielectric constant versus incidence angle calculated by two-phase mixture model. 68

Fig. 5-2 Power reflectivity at ice-bottom boundary versus incidence angle calculated by Fresnel's law. 70

Fig. 5-3. Backscattering coefficient at C-band (VV-polarization) versus each parameter characterizes tubular bubbles layer. 73-74

Fig. 5-4 Backscattering coefficient at the ice-bottom boundary versus incidence angle calculated by using the small perturbation model. 75

Fig. 5-5 Backscattering coefficient at C-band (VV) versus angle of incidence for the floating ice and the grounded ice..... 77

Fig. 5-6 Backscattering coefficient at C-band (VV) versus the thickness of an ice layer with tubular bubbles. 79

Fig. 5-7 The backscattering coefficient observed by ERS-1 SAR versus the backscattering coefficient estimated by lake ice model with specular ice-bottom boundary..... 81

Fig. 5-8 The backscattering coefficient observed by ERS-1 SAR versus the backscattering coefficient estimated by lake ice model with rough ice-bottom boundary..... 83

List of Tables

Table 2-1 Summary of snow and ice characteristics at each sampling site. 7

Table 2-2 Summary of ice core measurement 9-11

Table 3-1 List of C-band images used for this analysis. 21

Table 3-2. List of L-band images used for this analysis. 28

Table 3-3 The coordinates for sampling points on registered C-band image for each field observation point. 35

Table 3-4 The coordinates for sampling points on registered L-band image for each field observation point. 41

CHAPTER 1. INTRODUCTION

During the spring of 1974 a cooperative sea ice research program of the U.S. Geological Survey and the Cold Regions Research and Engineering Laboratory obtained X-band Side Looking Airborne Radar (SLAR) coverage of a portion of the coastal region of the North Slope of Alaska. This part of Alaska is very flat and is composed of tundra and a series of shallow lakes which comprise more than 40% of the surface area in many locations. During the winter, in areas where the snow pack is thin, as is commonly the case, lake ice thicknesses may exceed 2 meters, causing the shallower lakes to freeze completely to the bottom with the larger lakes remaining unfrozen only in their deeper parts.

An examination of the radar images of the lakes showed striking variations with the strengths of the microwave returns displaying two quite different primary modes: regions of very strong returns and regions of very low returns. When the images were shown to Paul Sellmann, an individual who had considerable experience in limnological studies of the lakes, he immediately suggested the possibility that the weak returns came from areas where the lakes were frozen to their beds while the strong returns indicated areas where water still remained beneath the ice. In two subsequent papers, Sellmann and coauthors [*Sellmann et al.*, 1975; *Weeks et al.*, 1977] described these observations and qualitatively explored possible explanations for the effect including surface roughness variations, thermal contrasts, ice thickness differences, and variations in ice/bottom interface conditions. They ruled out the first three explanations as improbable and suggested that the differences in return were in some way related to differences associated with the presence of an ice/water interface as contrasted with an ice/frozen soil interface. In the second of these papers they also examined the potential criticism that at X-band frequencies penetration would be insufficient to sense the conditions at the base of the ice. They concluded that for

fresh water ice attenuation distances were sufficiently large that this did not appear to be a limitation. Further observations [*Elachi et al.*, 1976] at L-band frequencies, obtained from large lakes in the vicinity of the Kuskokwim Delta near Bethel, Alaska, showed similar backscatter patterns. These results also indicated that the differences in returns were also obvious at the L-band wave lengths.

The first opportunity to obtain ground truth came in 1976 as a subsidiary project completed during research flights into the Beaufort Sea included as a part of the AIDJEX Project. The region imaged was located south of Lonely, Alaska and the coverage was more restricted than that obtained during the earlier flights. Both X- and L-band frequencies were used. The patterns of the returns were similar. The field checks [*Weeks et al.*, 1978] showed that, at all the sites where ice conditions were measured, strong returns were observed at locations where the lake ice was found not to be frozen to the bottom. Observations of the ice showed numerous, highly elongated air bubbles of sizes sufficient to cause scattering at either X- or L-band frequencies. In that the elongation of these bubbles was almost exactly orthogonal to the air-ice and ice-water interfaces, it was suggested that any forward scattering that occurred within the ice would result in the return of a significant portion of the electromagnetic energy at essentially the same angle as it was initially transmitted.

The final paper of this series [*Weeks et al.*, 1981] further expanded the ground-truth base for the interpretation of these radar patterns from 6 to 17 sites. The ubiquitous presence of elongated air bubbles was again noted and a clearer statement was given of the role of these bubbles as forward scatterers that ensure that a portion of the emitted radar pulse is returned to the receiver on the aircraft. It was suggested that the next step in the study of this general problem should be the mathematical modeling of the interactions between the radar pulse, the upper and lower ice surfaces, and the elongated bubbles enclosed in the ice. With the exception of an unpublished

Ph.D. thesis [*Mellor*, 1982], which utilized SLAR imagery to study North Slope lakes located somewhat to the south of the earlier sites and whose results were generally similar to the earlier studies, little work appears to have been on this phenomenon during the last 10 years. In particular no attempt has been made to develop a suitable simulation model to test the relative importance of the different hypothesized mechanisms. Presumably much of this neglect was associated with the very real lack of radar imagery during this period.

This situation is now drastically changed as the result of the launches of ERS-1 in July, 1991 and JERS-1 in February, 1992. The ERS-1, ESA (European Space Agency) satellite, has a C-band SAR system on board, while JERS-1, NASDA (National Space Develop Agency of Japan) satellite, has the L-band SAR system. Radiometric calibration of satellite-borne SAR data has been making good progress [*Fatland and Freeman*, 1992; *Shimada*, 1994], and geometric accuracy of SAR data has also been improved to achieve the same level as the high resolution optical sensor [*Fujimoto et al.*, 1989]. Both radiometrically and geometrically calibrated data, which are easy to use for general users, are now available and widely distributed. The SAR data from ERS-1 and JERS-1 satellites are acquired and processed at ASF (Alaska SAR Facility / University of Alaska Fairbanks) and EOC (Earth Observation Center / NASDA), and to date both stations have produced extensive coverage of the North Slope of Alaska including excellent sequential images of numerous ice covered lakes.

This paper presents field and laboratory measurement of lake ice, SAR data analysis, and the derivation of the first numerical model for lake ice. In CHAPTER 2, the result of field measurement and ice core measurement are described and lake ice characterization is discussed. Then ERS-1's and JERS-1's SAR data for lake ice are analyzed in CHAPTER 3. The numerical model for the lake ice is introduced and explained in CHAPTER 4. In CHAPTER 5, some simulations using this model are

demonstrated, and the modeled backscatter is compared with backscatter derived from C- band SAR images obtained at the same time as the field work. Finally, the conclusion is given in CHAPTER 6.

CHAPTER 2. LAKE ICE MEASUREMENT AND CHARACTERISTICS

In order to understand the lake ice characteristics and to make a numerical model for the lake ice, field measurements for the specific lakes are necessary. In April 1992, at the time lake ice had been grown, the field measurements of the lakes near Barrow, North West Alaska, were made [*Wakabayashi et al.*, 1992; *Jeffries et al.*, 1993, 1994]. This was the first field measurement for lake ice in northern Alaska done during almost simultaneous observation by a space borne SAR. During this field observation, ice cores were obtained and taken to the laboratory for detailed ice core analysis.

In this chapter, the results of field measurements are described, such as snow depth on ice, ice thickness, ice bottom interface and so on, then the results of the ice core analysis such as layer characteristics, ice thickness of each layer, porosity, and bubble shape are shown. Finally, lake ice characterization is discussed.

2-1. TEST SITE

The lakes chosen for the field observation were located in the vicinity of Barrow, the northernmost community in Alaska. This site was selected for the reasons which follow;

(1) It is easy to access from Fairbanks by airplane.

(2) Previous studies [*Sellman et al.*, 1975; *Weeks et al.*, 1977] indicated that the lakes near Barrow showed large variations in radar backscatter.

(3) The ERS-1 took a three days cycle orbit in the first stage after its launch, during the period from July 1991 to March 1992. In this period, Barrow data could be acquired every three days.

(4) We have a person who knows about the Barrow area a lot. Since the area around Barrow is very flat and covered with snow in winter time, the site becomes fairly featureless. In order to find the specific lakes, a knowledgeable person is necessary. Also we used GPS receivers to check the locations as a double check.

During April 1992, 11 locations for field observation were picked up on the ERS-1's SAR imagery to cover various backscatters; some of which showed strong backscatter, some low ones. The locations of 11 sites on the map are shown in Fig. 2-1.

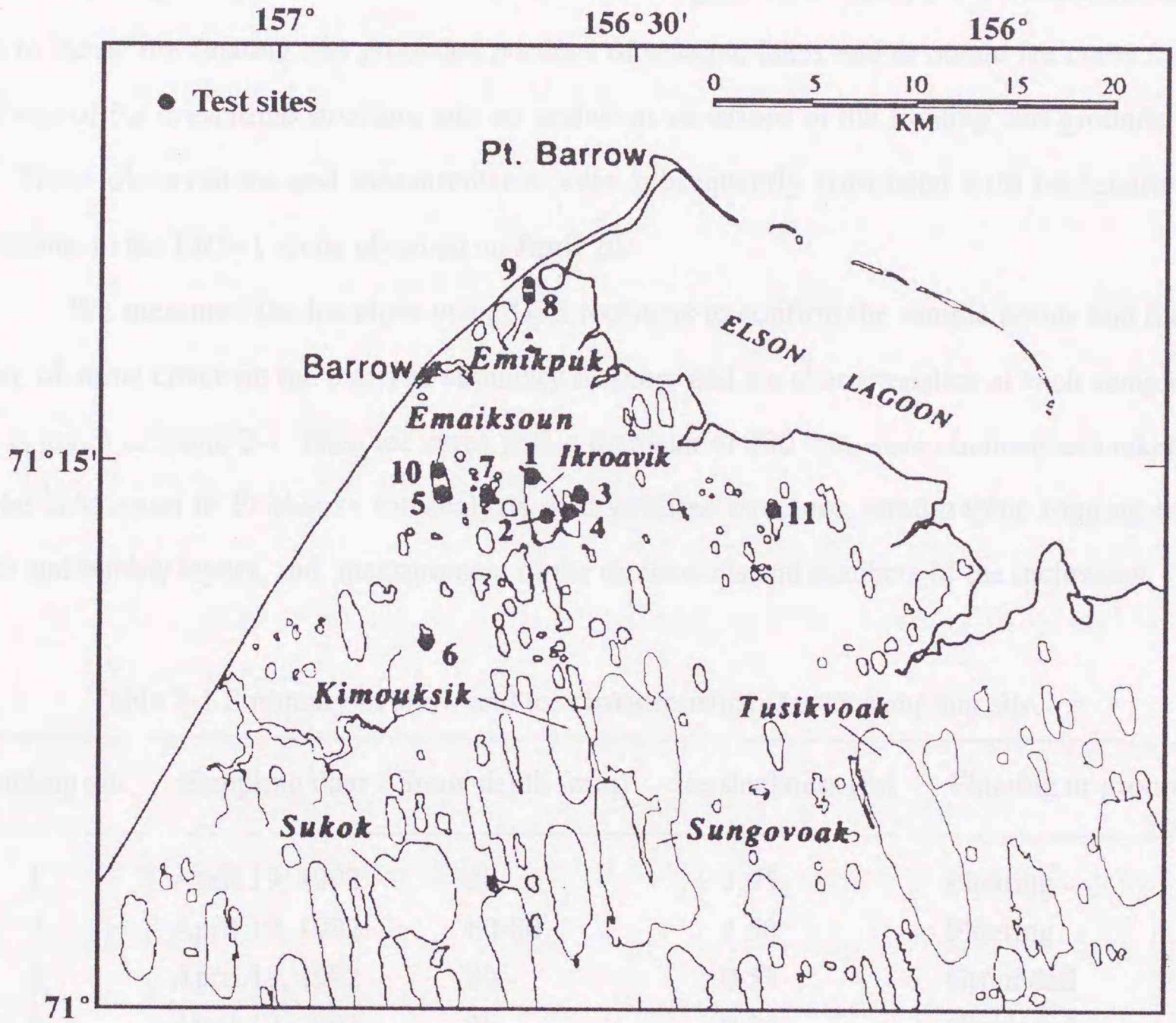


Fig. 2-1 Map of the region around Barrow including field observation locations. The numbers on the map stand for the sampling points of field measurement.

2-2. FIELD MEASURED DATA

Field work was undertaken near Barrow in late April 1992. Ice core samples were obtained from sites 1-8 (Fig. 2-1) on April 19-20 concurrent with an ERS–1 overpass on April 20. A few days later, April 28-29, three additional sites (9-11) were sampled. Changes in the ice cover during this period are believed to have been negligible. The purpose of the fieldwork was to locate the floating and grounded portions of selected lakes and to obtain ice cores for analysis of ice crystalline structure and air inclusion variations in the floating and grounded ice. These observations and measurements were subsequently correlated with backscatter variations in the ERS–1 scene obtained on April 20.

We measured the locations using GPS receivers to confirm the sample points and the depth of snow cover on the ice. The summary of snow and ice characteristics at each sample site is given in Table 2-1. Then ice cores with a diameter of 100 mm were obtained and taken to the cold room in Fairbanks for analysis of crystalline structure, stratigraphic logging of clear and bubbly layers, and measurement of the dimensions and numbers of the inclusions.

Table 2-1 Summary of snow and ice characteristics at each sampling site.

Sampling site	Sampling date	Snow depth (mm)	Ice thickness (m)	Floating or grounded
1	April 19, 1992	30	1.95	Floating
2	April 19, 1992	60-80	1.59	Floating
3	April 19, 1992	30	0.53	Grounded
4	April 19, 1992	26	0.24	Grounded
5	April 20, 1992	0-50	2.10	Floating
6	April 20, 1992	80	1.18	Grounded
7	April 20, 1992	20	0.52	Grounded
8	April 20, 1992	50	2.01	Floating
9	April 28, 1992	25	2.16	Floating
10	April 29, 1992	60	2.04	Floating
11	April 29, 1992	30	0.77	Grounded

As shown in Table 2-1, ice thickness varied between 0.24 m and 2.16 m. The ice was floating at six sampling sites and was frozen to the lake bottom at five others. The snow cover was dry and packed hard in most cases, probably by wind action. Occasionally, the wind had blown the snow completely away, exposing patches of bare ice.

2-3. ICE CORE MEASUREMENT

Ice cores were obtained from 10 sites and carried to the cold room in the Geophysical Institute / University of Alaska Fairbanks. First each ice core was divided into horizontal layers having the same bubble densities and bubble shapes, then thickness of layer and inclusion characteristics were measured. The summary of ice core measurement is given in Table 2-2. As the consequence of this measure, it was found that ice cores contained three particular types of ice:

(1) granular ice containing roughly spherical bubbles;

(2) inclusion–free ice, some with the columnar texture indicating crystals with horizontal c–axes, and some with the massive, single crystal structure typical of ice with vertically–oriented c–axes;

(3) columnar and massive single ice crystals containing tubular (cylindrical) bubbles oriented roughly perpendicular to the ice surface, i.e. parallel to the ice growth direction.

The granular ice occurred at the top of cores from six locations (sites 2, 3, 5, 6, 7 and 11). The clear ice occurred at all sites, either as the uppermost layer or directly below the granular ice . Ice with tubular bubbles occurred as the lowermost layer at all locations.

The thickness of the granular layers varied between 0.01 m and 0.22 m. The inclusion–free layer thicknesses ranged from 0.14 m to 1.18 m. The depth below the surface at which tubular bubbles appeared varied from 0.26 m to 1.18 m. Tubular bubbles appeared earliest at the shallowest lake sites where the ice was frozen to the bottom when the samples were obtained. The thickness of the ice layers with tubular bubbles ranged from 0.27 m to 1.32 m. The illustrated structure and ice type distribution for each ice core is shown schematically in Fig. 2-2 and sample photographs for sub-layers are shown in Fig. 2-3.

Table 2-2 Summary of ice core measurement (1/3).

Site	Characteristics	Depth (m)	Thickness (cm)	Number of bubbles (count/cm^2)	Bubble diameter (mm)	Porosity (%)	Mean bubble length (cm)
1	Clear	0.00	62.0	0.0	1.2	0.0	
	Tubular	0.62	9.0	0.4	0.8	0.2	3.89
		0.71	10.0	7.0	0.6	2.0	
		0.81	7.0	10.0	0.7	3.8	
		0.88	15.0	9.0	0.7	3.5	
		1.03	74.5	34.0	0.4	4.3	
		1.78	7.0	7.0	0.8	3.5	
		1.85	10.0	3.7	0.8	1.8	
2	Granular	0.00	1.0	1.0	0.7	0.0	0.07
	Clear	0.01	30.0	0.0	0.9	0.0	
	Tubular	0.31	10.0	0.0	0.0	0.0	9.10
		0.41	12.0	1.0	1.8	2.5	
		0.53	16.0	0.0	1.8	0.0	
		0.69	13.0	0.0	2.4	0.0	
		0.82	7.0	2.0	1.7	4.5	
		0.90	9.0	2.0	1.4	3.1	
		0.99	22.0	2.0	1.5	3.5	
		1.21	16.0	1.0	1.7	2.3	
		1.37	8.0	1.0	1.9	2.8	
		1.45	9.0	1.0	1.9	2.8	
		1.54	5.0	0.0	0.0	0.0	
3	Granular	0.00	12.0	3.0	4.6	4.3	0.46
	Clear	0.12	14.0	0.0	0.0	0.0	
	Tubular	0.26	4.0	12.0	1.1	11.4	2.90
		0.30	2.0	5.0	0.5	1.0	
		0.32	7.0	14.0	0.5	2.7	
		0.39	14.0	14.0	0.3	1.0	

Table 2-2 Summary of ice core measurement (2/3).

Site	Characteristics	Depth	Thickness	Number of bubbles	Bubble diameter	Porosity	Mean bubble length
		(m)	(cm)	(count/cm^2)	(mm)	(%)	(cm)
5	Granular	0.00	22.0	4.0	2.9	3.1	0.29
	Clear	0.22	74.0	0.0	0.0	0.0	
	Tubular	0.96	13.5	1.0	0.7	0.4	4.77
		1.10	8.5	2.0	0.6	0.6	
		1.18	7.0	2.0	0.9	1.3	
		1.25	9.5	2.0	0.7	0.8	
		1.35	4.5	3.0	0.6	0.8	
		1.39	6.5	4.0	0.4	0.5	
		1.46	6.5	14.0	0.7	5.4	
		1.52	25.0	3.0	0.5	0.6	
		1.77	8.0	26.0	0.5	5.1	
		1.85	15.0	5.0	0.8	2.5	
		2.00	10.0	5.0	1.2	5.7	
6	Granular	0.00	3.0	1.0	3.0	1.6	0.30
	Clear	0.03	41.0	0.0	0.0	0.0	
	Tubular	0.44	13.0	0.0	1.0	0.0	4.44
		0.57	6.0	1.0	0.8	0.5	
		0.63	15.0	3.0	0.6	0.8	
		0.78	14.0	8.0	0.3	0.6	
		0.92	7.0	15.0	0.5	2.9	
		0.99	19.0	15.0	0.6	4.2	
7	Granular	0.00	6.0	3.0	2.5	0.6	0.25
	Clear	0.06	25.0	0.0	0.0	0.0	
	Tubular	0.31	8.0	16.0	0.4	2.0	2.90
		0.39	2.0	13.0	0.3	0.9	
		0.41	11.0	11.0	0.5	2.2	
8	Clear	0.00	82.0	0.0	0.0	0.0	
	Tubular	0.82	32.0	2.0	0.5	0.4	2.77
		1.14	4.0	4.0	0.8	2.0	
		1.18	23.0	1.0	0.6	0.3	
		1.41	9.0	3.0	0.5	0.6	
		1.50	4.0	2.0	0.5	0.4	
		1.54	29.0	1.0	0.6	0.3	
		1.83	12.0	4.0	0.6	1.1	
		1.95	5.5	1.0	0.8	0.5	

Table 2-2 Summary of ice core measurement (3/3).

Site	Characteristics	Depth (m)	Thickness (cm)	Number of bubbles (count/cm^2)	Bubble diameter (mm)	Porosity (%)	Mean bubble length (cm)
9	Clear	0.00	118.0	0.0	0.0	0.0	
	Tubular	1.18	2.0	11.0	0.5	2.2	2.88
		1.20	20.0	10.0	0.6	2.8	
		1.40	14.0	9.0	0.5	1.8	
		1.54	18.0	13.0	0.4	1.6	
		1.72	15.0	15.0	0.5	2.9	
		1.87	12.0	10.0	0.6	2.8	
		1.99	16.0	2.0	1.0	1.6	
10	Clear	0.00	98.0	0.0	0.0	0.0	
	Tubular	0.98	3.5	4.0	0.7	1.5	5.31
		1.02	19.0	3.0	0.6	0.8	
		1.21	5.0	3.0	0.7	1.2	
		1.26	19.5	3.0	0.6	0.8	
		1.45	16.0	20.0	0.5	3.9	
		1.61	17.5	14.0	0.8	7.0	
		1.79	9.0	5.0	1.2	5.7	
		1.88	16.5	5.0	0.9	3.2	
11	Granular	0.00	5.0	1.0	2.5	0.1	0.25
	Clear	0.05	33.0	0.0	0.0	0.0	
	Tubular	0.38	5.0	6.0	0.6	1.7	1.48
		0.43	6.0	41.0	0.4	5.2	
		0.49	19.0	29.0	0.3	2.0	
		0.68	9.0	28.0	0.3	2.0	

In the surface granular layers, the mean bubble diameter ranged from 0.7 mm at site-2 to 4.6 mm at site-3, and the number of bubbles varied between 10.8 x 10^3 /m^3 to 2,471 x 10^3 /m^3. The density of tubular bubbles at the base of the ice varied between 0.4 x 10^3 /m^2 to 411 x 10^3 /m^2. The radius of the cross section of the tubular bubbles ranged from 0.15 mm to 1.2 mm. The largest diameter bubbles were found only at site-2; otherwise, the majority of tubular bubbles had a small radius averaging 0.4 mm at the other sites. The mean length of tubular bubbles ranged from 1.4 cm at Site-11 to 9.1 cm at Site-2. The shortest and longest bubbles had lengths of 0.7 cm and 17.8 cm respectively. At any given sampling site, the density and mean diameter of the tubular bubbles varied with depth. Consequently, the ice with tubular bubbles comprised a number of sub–layers illustrated in Fig. 2-2. The porosity (volume inclusion factor) profile for each test site is shown in Fig. 2-4. The porosity n_p is determined according to the simple relation as follows,

$$n_p = \frac{\rho_i - \rho_s}{\rho_i}$$

where ρ_i and ρ_s are the densities of pure bubble-free ice and of the ice sample. Only one case for the porosity over 10 % was found at site-3 with mean bubble diameter 1.1 mm and mean bubble length 2.9 cm. For most case, the porosity was found within 6 %.

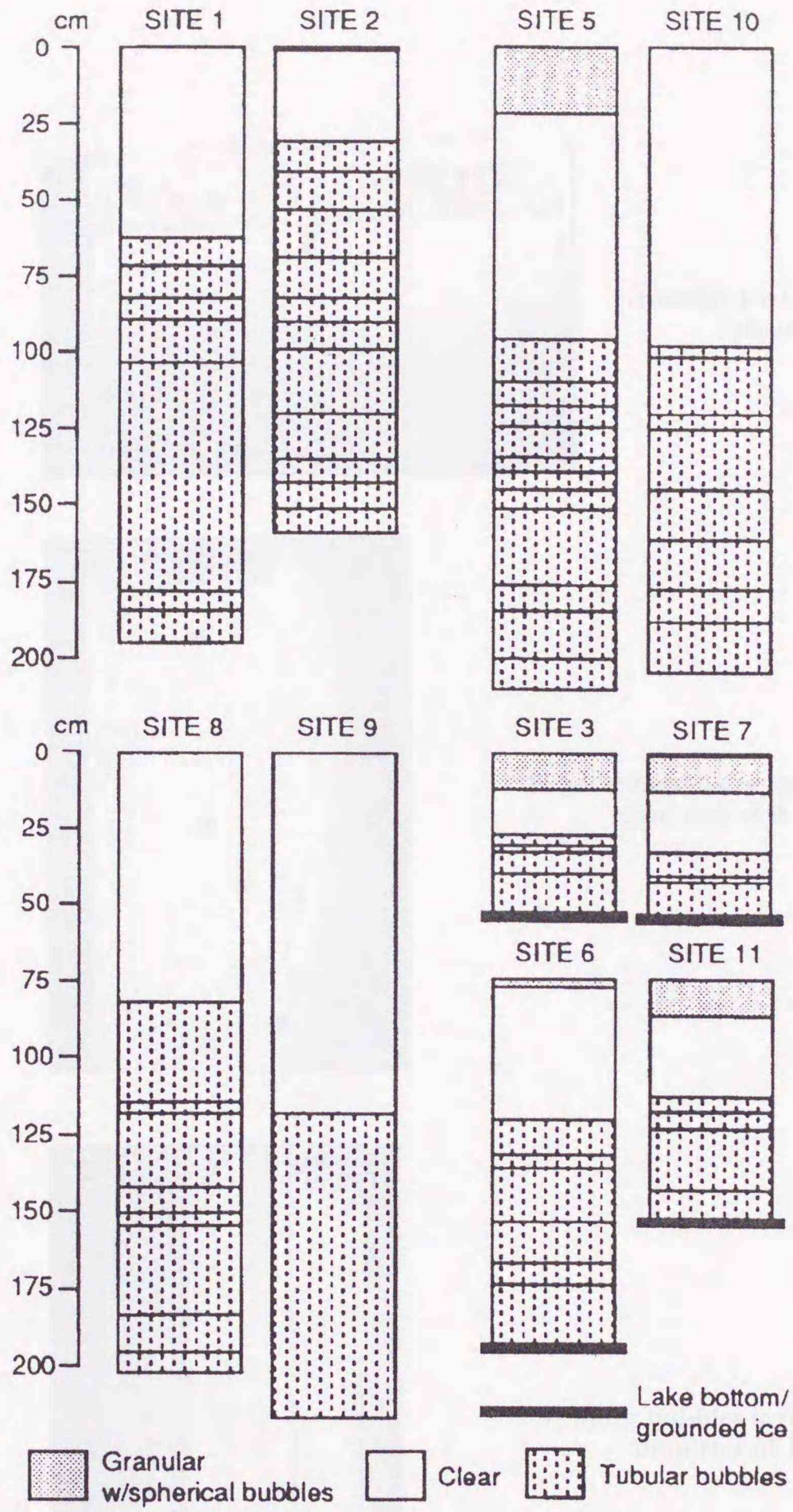
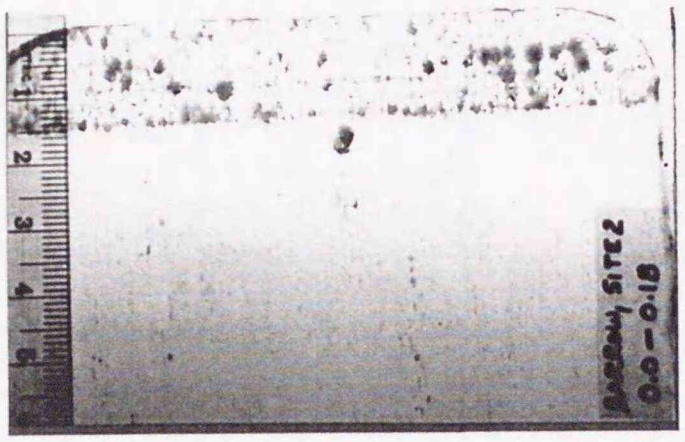
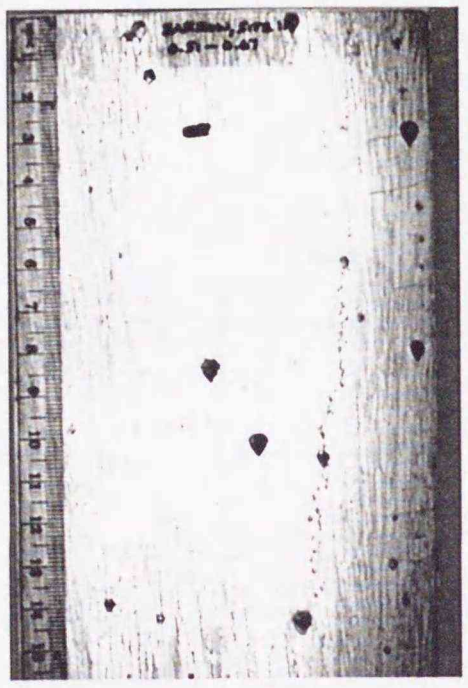


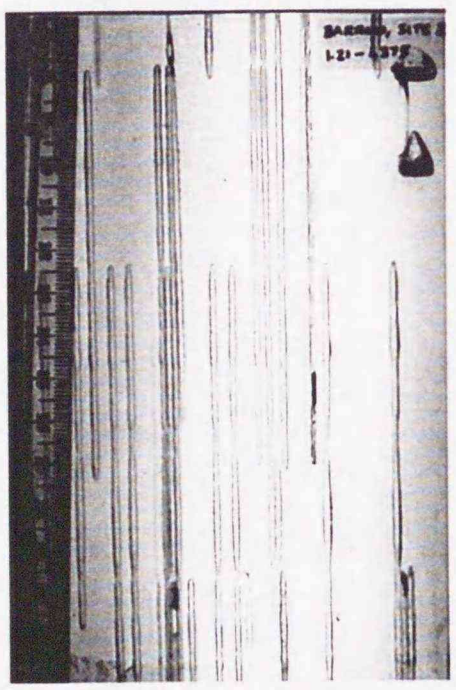
Fig. 2-2 Diagrams for ice cores illustrating the amount and distribution of ice types.



Granular ice layer with a mean bubble diameter of 0.7 mm



Tubular bubbles layer with a mean diameter of 0.3 mm



Tubular bubbles layer with a mean diameter of 1.7 mm

Fig. 2-3 Sample photographs for ice cores in sub layers

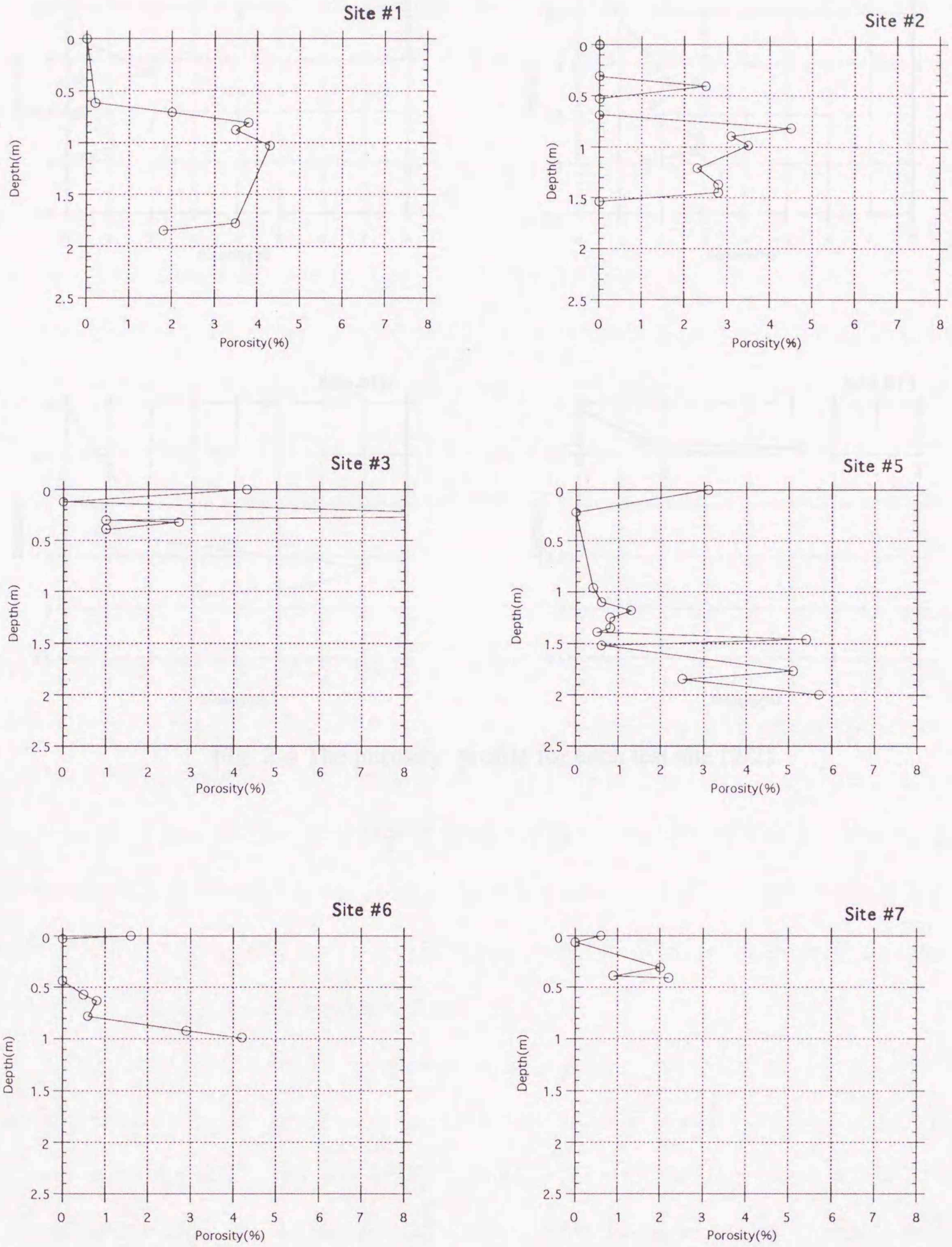
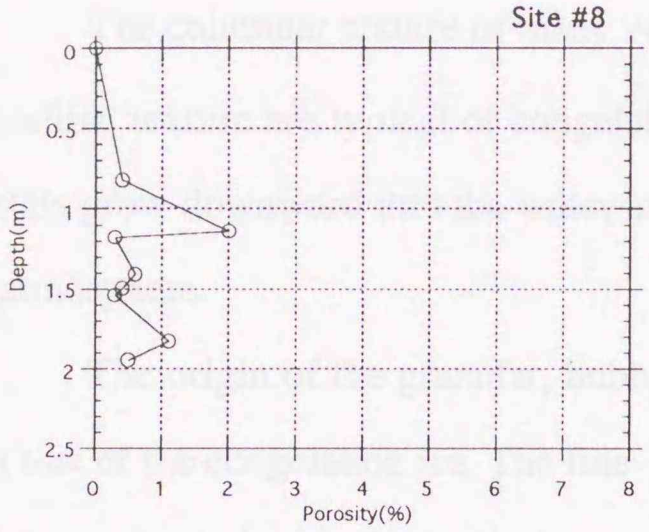
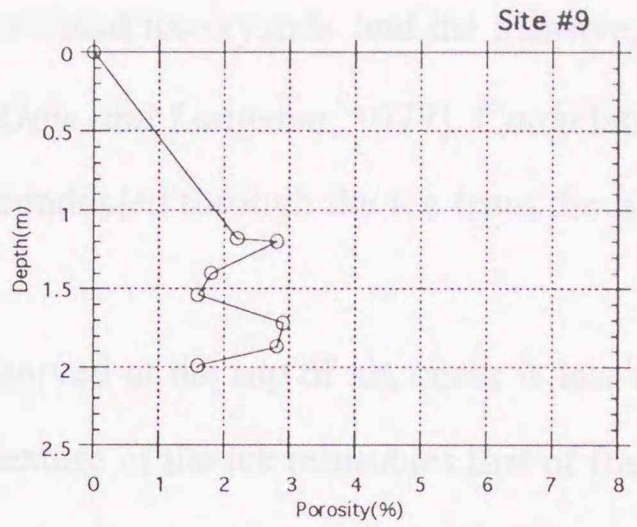
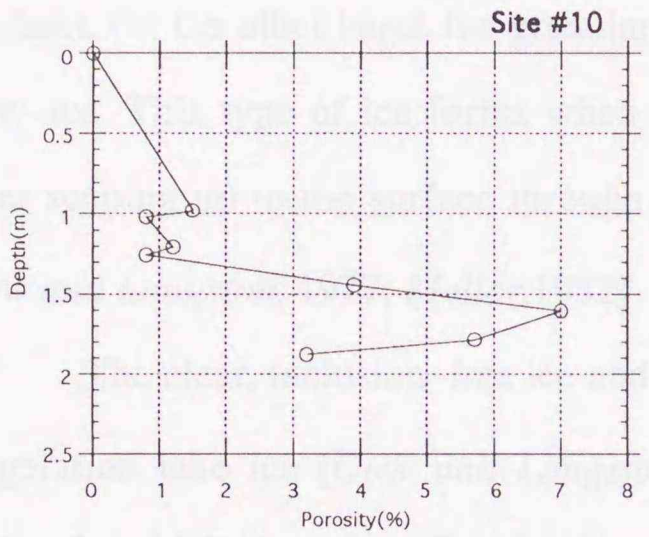


Fig. 2-4 The porosity profile for each test site (1/2).







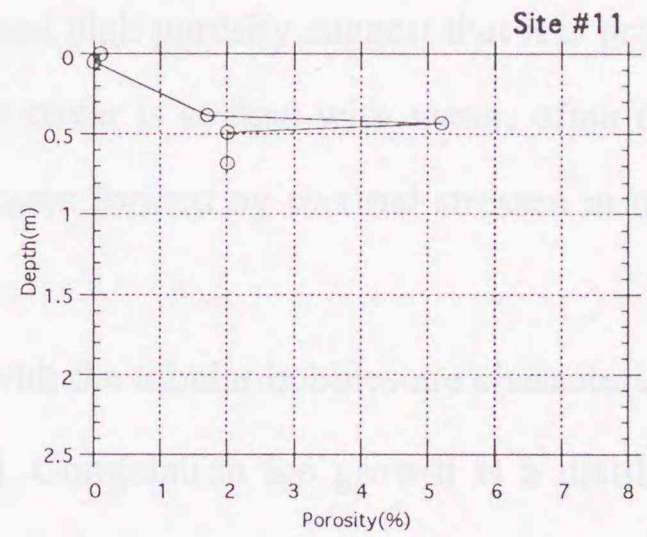


Fig. 2-4 The porosity profile for each test site (2/2).

2-4. ICE CHARACTERIZATION

The columnar texture of many vertically oriented ice crystals and the massive, single crystalline texture are typical of congelation ice [*Gow and Langston*, 1977]. Congelation ice crystals grow downward into the water as heat is conducted through the ice from the water to the atmosphere.

The origin of the granular, bubbly ice observed at the top of six cores is less certain than that of the congelation ice. The fine–grained texture of the ice resembles that of frazil ice, which can form during the earliest stages of ice growth when the water surface is turbulent and newly–formed ice crystals are broken into small pieces before consolidating into a solid ice sheet. On the other hand, the granular texture and high porosity suggest that it is probably snow–ice. This type of ice forms when the snow cover is soaked with water, often due to water seeping up to the surface through the fractures formed by thermal stresses in the ice [*Gow and Langston*, 1977; *Mellor*,1982]

The clear, inclusion–free ice and the ice with the tubular bubbles are characteristic of congelation lake ice [*Gow and Langston*, 1977]. Congelation ice growth is a distillation process in which impurities dissolved in the water, including gases, are rejected back into the water from the growing ice. The gases remain in solution and are not included in the ice unless the rate of gas rejection, which is proportional to the ice growth rate, is rapid enough to cause the water's gas content at the growing interface to exceed its nucleation level [*Gow and Langston,* 1977] and/or the water becomes supersaturated in gases for reasons that are independent of the ice growth rate [*Bari and Hallett,* 1974].

The clear ice observed in the Barrow cores represents the earlier stages of congelation ice development when all gases were rejected back into the water. Later in the ice growth season on these shallow lakes, it is likely that the water becomes supersaturated with gases and the tubular bubbles are the characteristic means by which gases are subsequently included in the ice. As the shallower lakes are a smaller reservoir for the gases rejected during ice growth, they commonly become supersaturated with gas earlier than the deeper lakes; hence the observation that the top of the tubular bubble layer is at a different level at each site, with

the tubular bubbles appearing first in the shallower lakes with the thinner, grounded ice covers. Once tubular bubble development is underway, the variations in bubble density, length and radius may be affected by variations in the ice growth rate and the gas concentration in the water [*Bari and Hallett*, 1974; *Gow and Langston*, 1977].

CHAPTER 3. SAR DATA FOR LAKE ICE

In August 1991, the ASF in Fairbanks began to receive and process data from C-band SAR aboard the ERS-1. After a calibration and validation campaign during the summer and autumn in 1991, the radiometrically calibrated SAR data were released to the scientific community. From September 1991 to April 1992, a time-series of scenes of the Barrow region (Fig. 2-1) were acquired and processed.

The ASF also started receiving L-band SAR data around Alaska from JERS-1 in mid 1992. The acquired raw data were transported to the EOC every two weeks and processed to make radiometrically calibrated images. We obtained 7 JERS-1/SAR images for Barrow area from July 1992 to April 1993.

In this chapter, spatial and temporal variations in backscatter from ice growing on a number of lakes are derived from the ERS-1 and the JERS-1 SAR scenes which are radiometrically calibrated. Then the variation of SAR data for the specific lakes are extracted and the relation between backscatter and ice-bottom interface is investigated. Using the radiometrically calibrated SAR data, the change of backscattering coefficient associated with ice-bottom interface and growing ice is confirmed for the first time.

3-1. SAR DATA

The SAR is an active microwave sensor and can acquire images on the earth in most weather conditions. The resolution of SAR data is much higher than the other microwave sensors such as a microwave radiometer [*Maeda et al.*, 1987; *Cho et al.*, 1989]. Although the resolution of the current space borne SAR is the order of 10 meter, the high resolution data are suffering from a nuisance factor which is called speckle noise. In order to get the calibrated backscattering coefficient (σ^0) from lake ice imagery, it is necessary to apply the speckle noise reduction technique. Several speckle reduction techniques have been proposed, which can effectively reduce noise as well as preserve

structured targets using local statistics [*Wakabayashi and Arai*, 1995a, 1996a, 1996b]. However, since the target lakes are much larger than the resolution of SAR data, simple averaging is enough to get accurate backscattering coefficient.

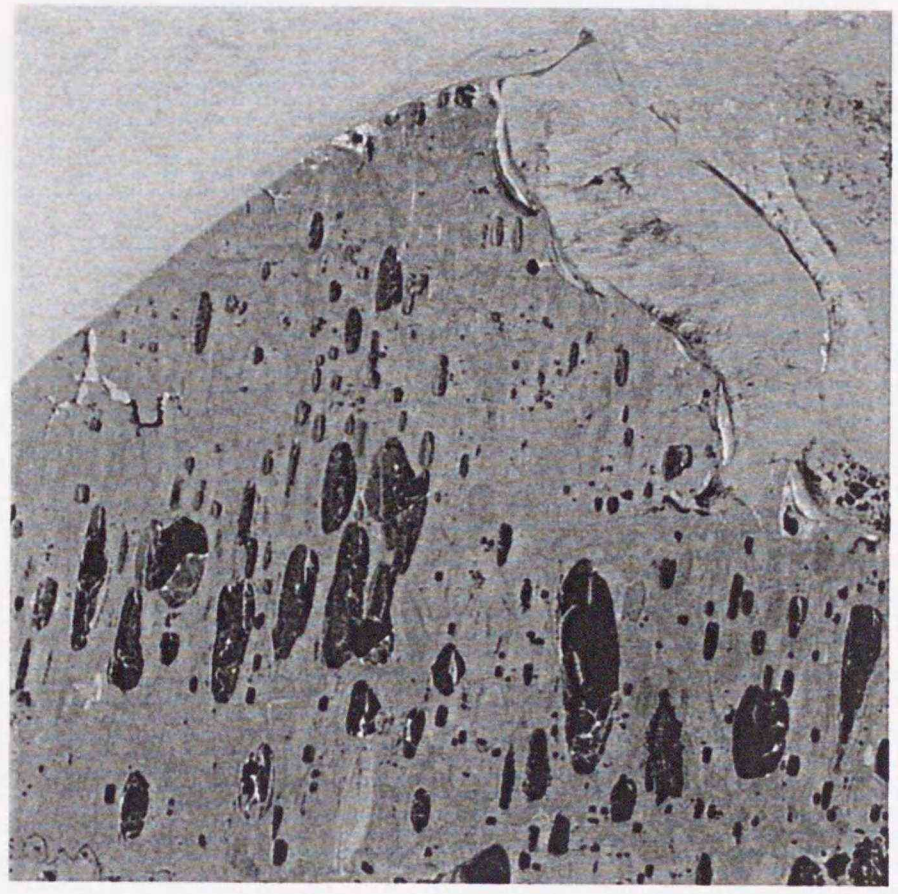
3-1.1. C-BAND SAR DATA

In order to use SAR imagery to observe changes in backscatter including the time change, the total of 12 scenes of ERS-1 low-resolution scenes (100 m pixel size which is made by averaging 8 x 8 pixels of full resolution data) were obtained and processed at the ASF station during the winter of 1991-92 [*Wakabayashi et al.*, 1992]. Table 3-1 shows the list of images, including date of acquisition, image ID, and acquisition mode (ascending or descending). The first scene was taken in late September 1991 when the lake ice cover was first established, and the last scene was obtained in late April 1992 when the lake ice had grown over 2 m.

As shown in Table 3-1, the ERS-1 SAR data were taken in different acquisition modes, some of them were acquired in an ascending mode and the others were in a descending mode. First, these data should be registered to the reference coordinate and then compared with each other. The extracted image from the image ID 8989200 was chosen as a reference coordinate, which has a size 512 pixels by 512 lines, and the other 11 images were registered to this reference using first order transformation with at least 8 control points. Since the geometric distortion caused by terrain effect is considered to be very small because of the flatness of the chosen area, the first order transformation is considered enough for this registration. The Digital Number (DN) value in the registered images were converted into σ^0 related value according to the calibration equation [*Bicknell*, 1992]. The time-series images which are registered and calibrated are given in Fig. 3-1.

Table 3-1 List of C-band images used for this analysis.

No	Image ID	Acquisition Date	Julian Day	Ascending/ Descending
1	8989200	9/29/91	272/91	A
2	4884200	10/17/91	290/91	A
3	1068200	10/26/91	299/91	A
4	6145200	12/01/91	335/91	A
5	8913200	12/30/91	364/91	D
6	9548200	1/14/92	014/92	D
7	11596200	1/29/92	029/92	D
8	10304200	2/10/92	041/92	D
9	11083200	2/25/92	056/92	D
10	15026200	3/11/92	071/92	D
11	16248200	3/29/92	089/92	D
12	18465200	4/20/92	111/92	D

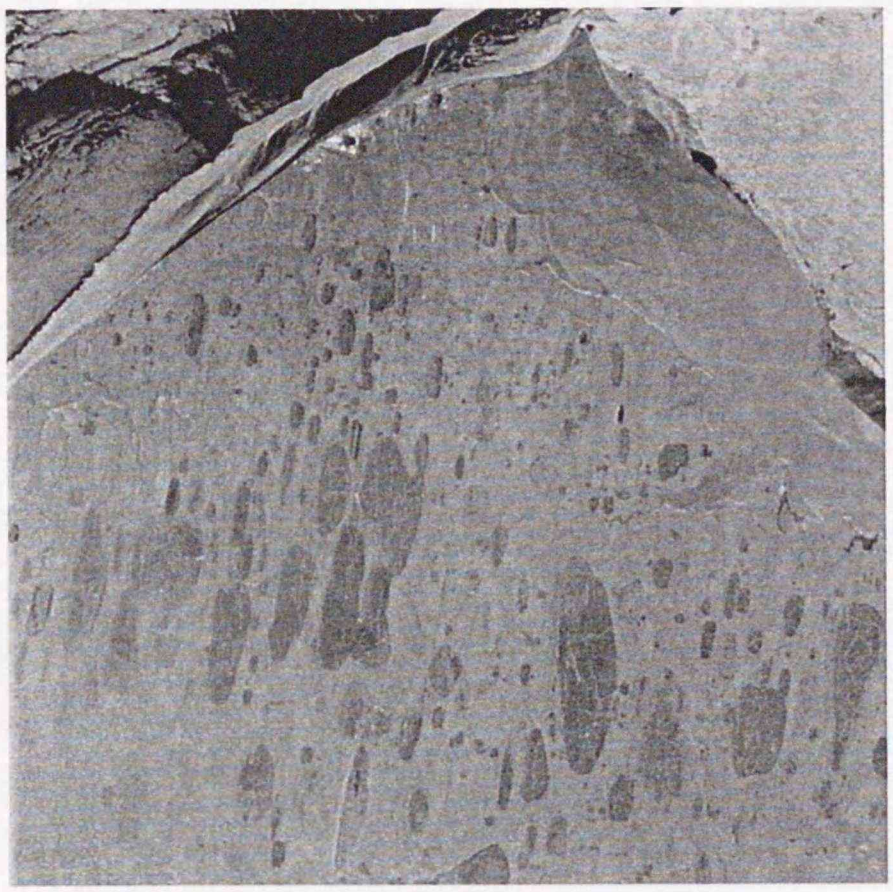


(1) ID:8989200 (9/29/91)

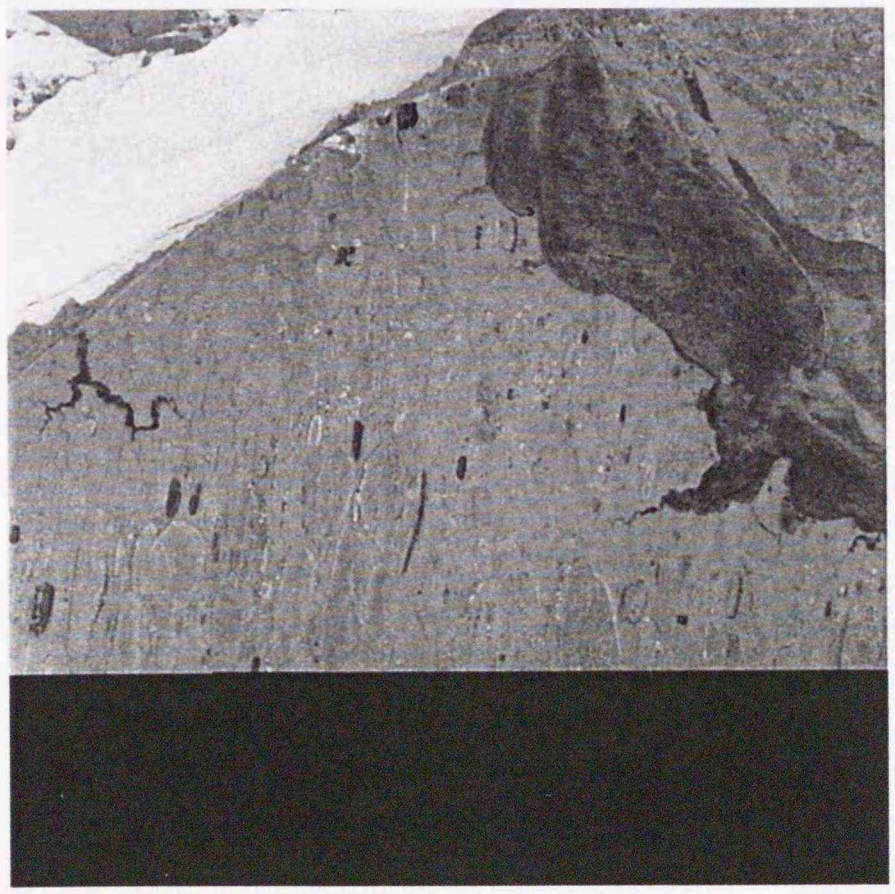


(2) ID:4884200 (10/17/91)

Fig. 3-1 Time series of C-Band radar images for lakes near Barrow(1/6).



(3) ID:1068200 (10/26/91)

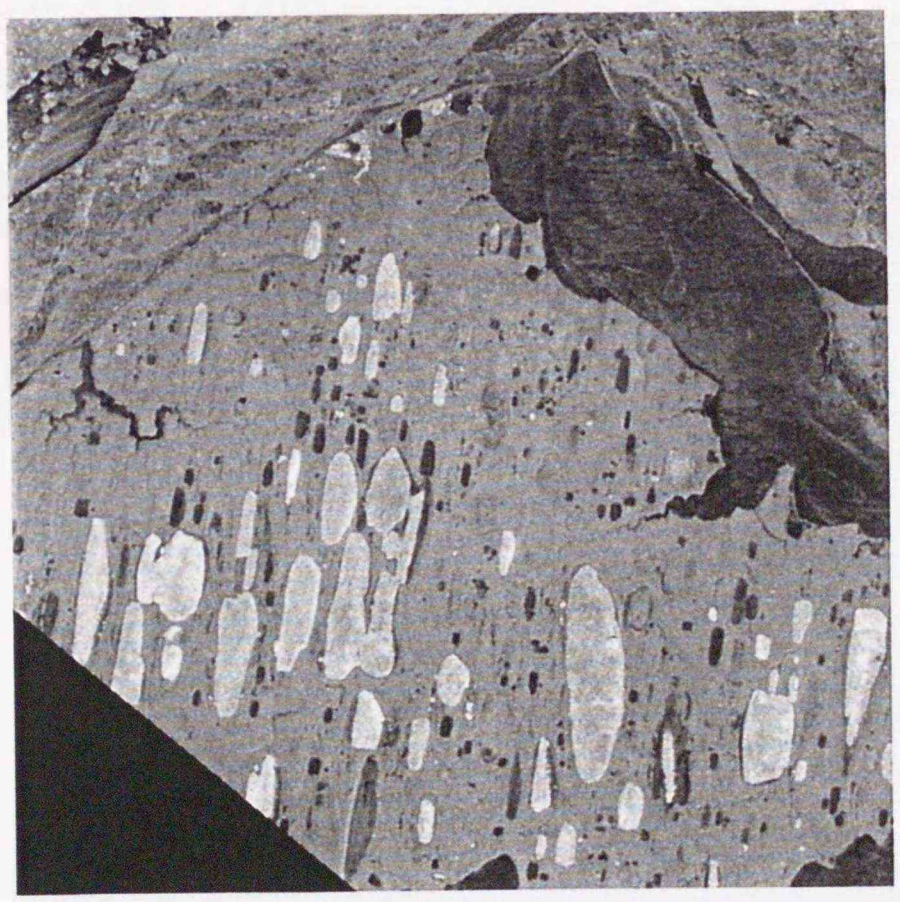


(4) ID:6145200 (12/01/91)

Fig. 3-1 Time series of C-Band radar images for lakes near Barrow(2/6).

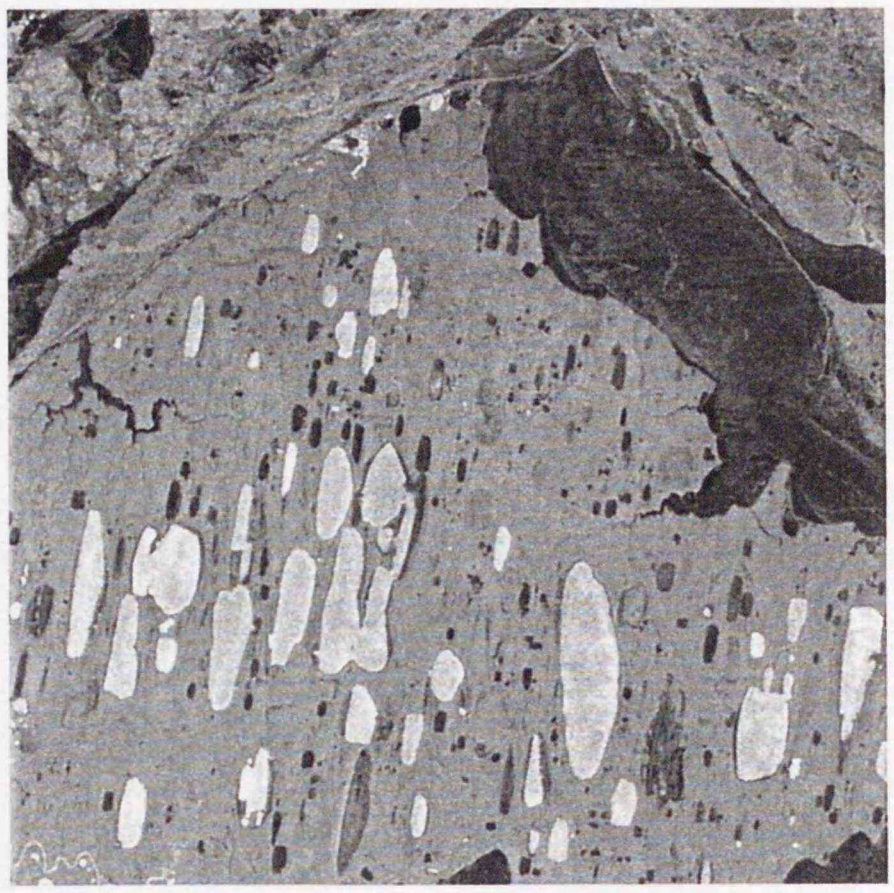


(5) ID:8913200 (12/30/91)

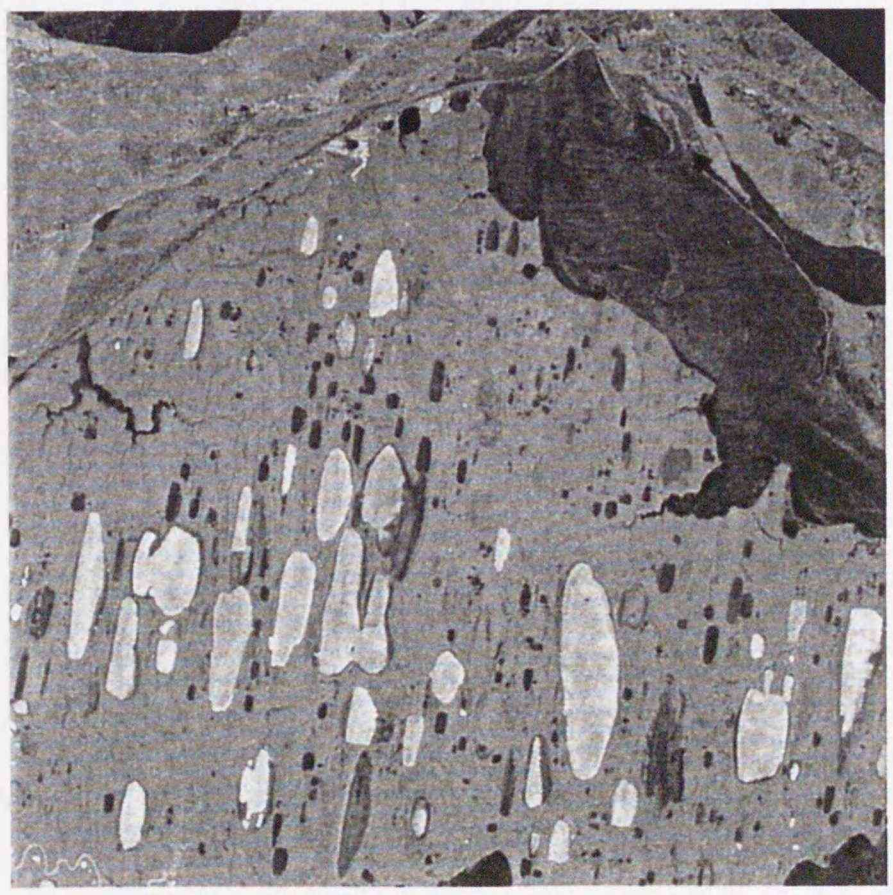


(6) ID:9548200 (1/14/92)

Fig. 3-1 Time series of C-Band radar images for lakes near Barrow(3/6).



(7) ID:11596200 (1/29/92)



(8) ID:10304200 (2/10/92)

Fig. 3-1 Time series of C-Band radar images for lakes near Barrow(4/6).

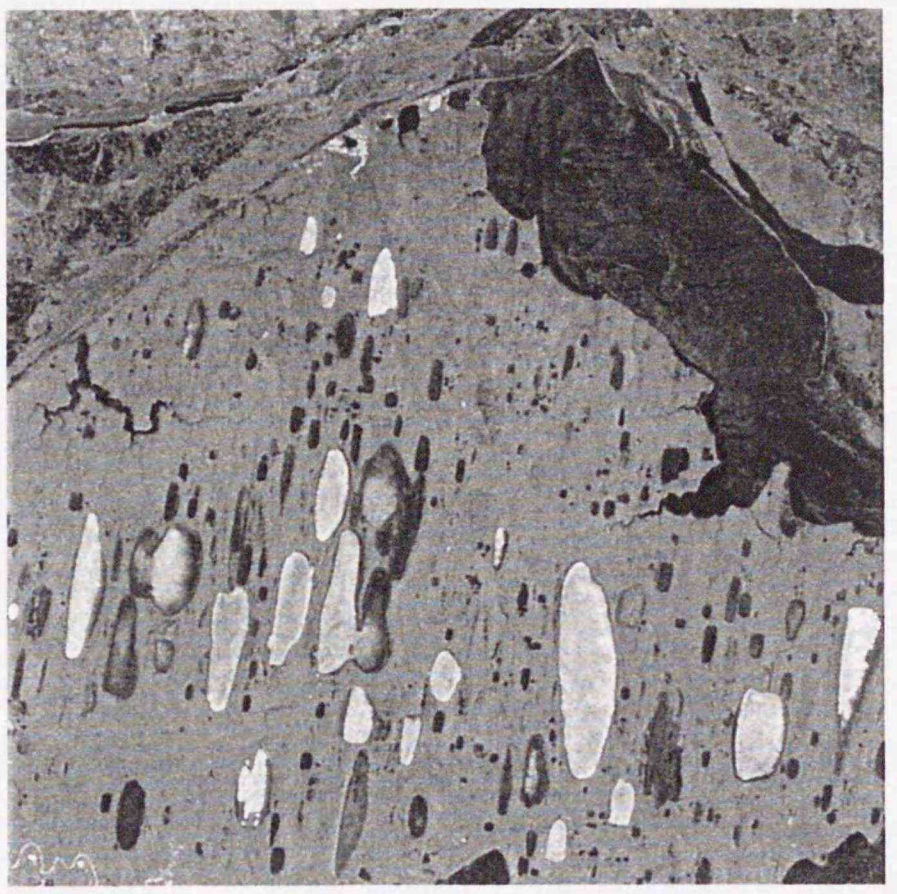


(9) ID:11083200 (2/25/92)

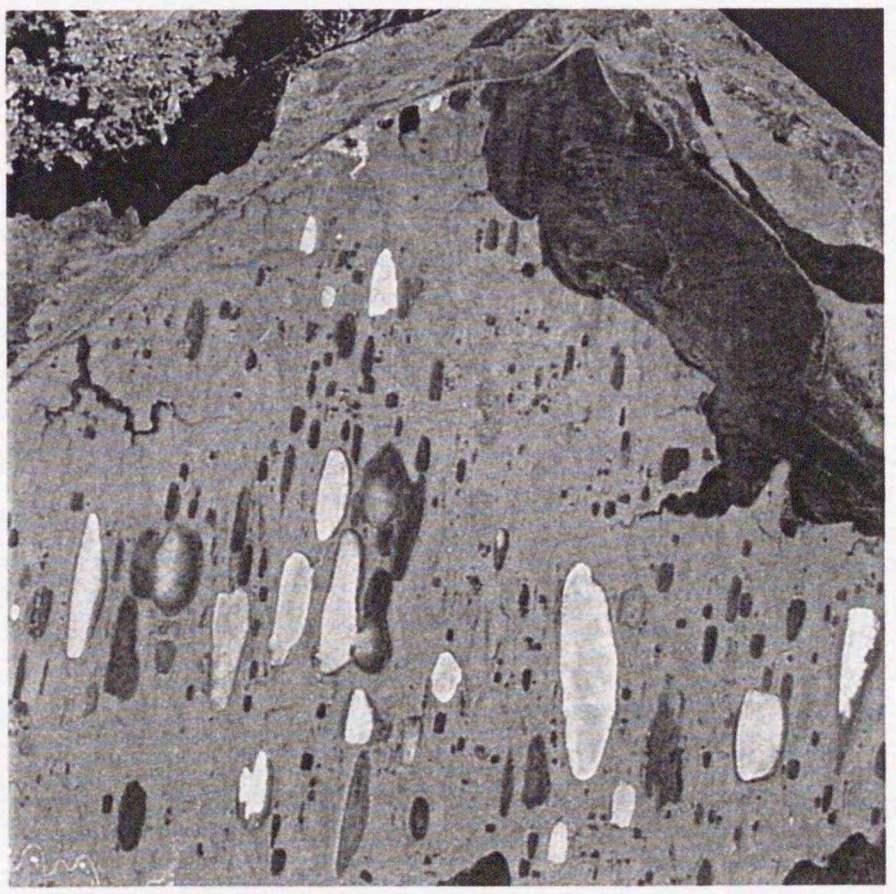


(10) ID:15026200 (3/11/92)

Fig. 3-1 Time series of C-Band radar images for lakes near Barrow(5/6).



(11) ID:16248200 (3/29/92)



(12) ID:18465200 (4/20/92)

Fig. 3-1 Time series of C-Band radar images for lakes near Barrow(6/6).

3-1.2. L-BAND SAR DATA

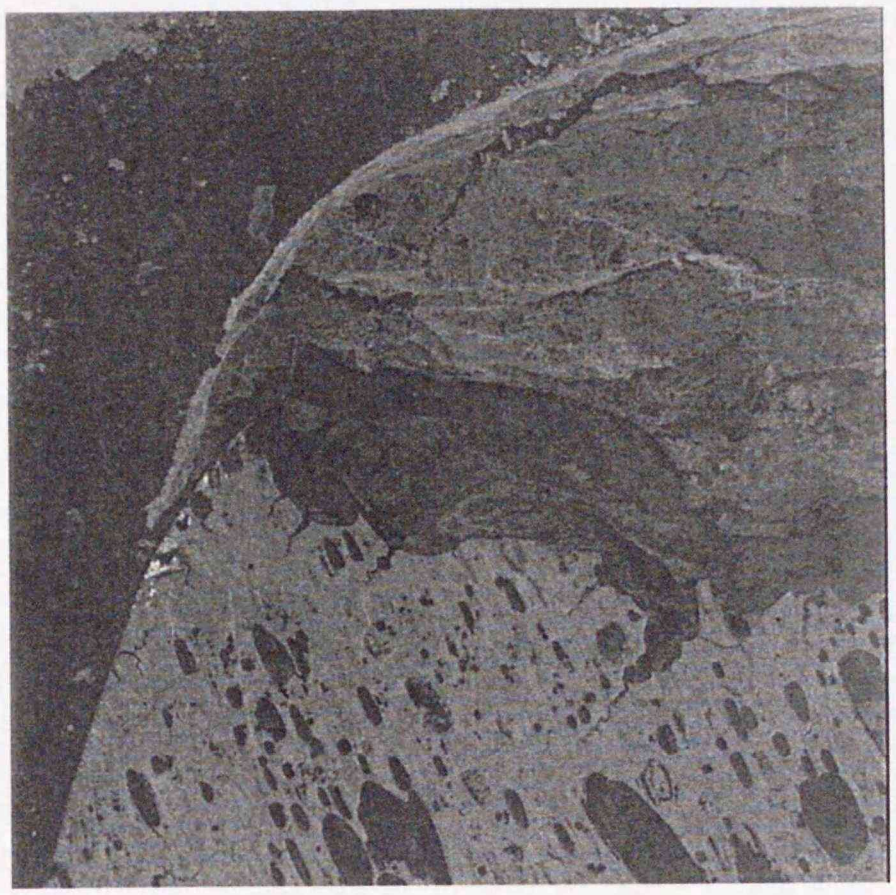
The JERS-1 was launched from Tanegashima island on 11 February 1992, and its L-Band SAR has taken a lot of images worldwide using many ground stations and its on-board recorder (MDR). Because of mission planning constraints and the orbit recurrent setting, which is a 44 days cycle, we had a few opportunities to acquire data from the area of interest. From July 1992 to April 1993, there were several opportunities to take images near Barrow area by using the on-board recorder or real time receiving of the ASF. After transporting data from the ASF to the EOC, the data (Image location ID Path: 645 / Row: 180) were processed at EOC station in a standard processing mode. The list of the acquired data is shown in Table. 3-2.

Table 3-2. List of L-band images used for this analysis.

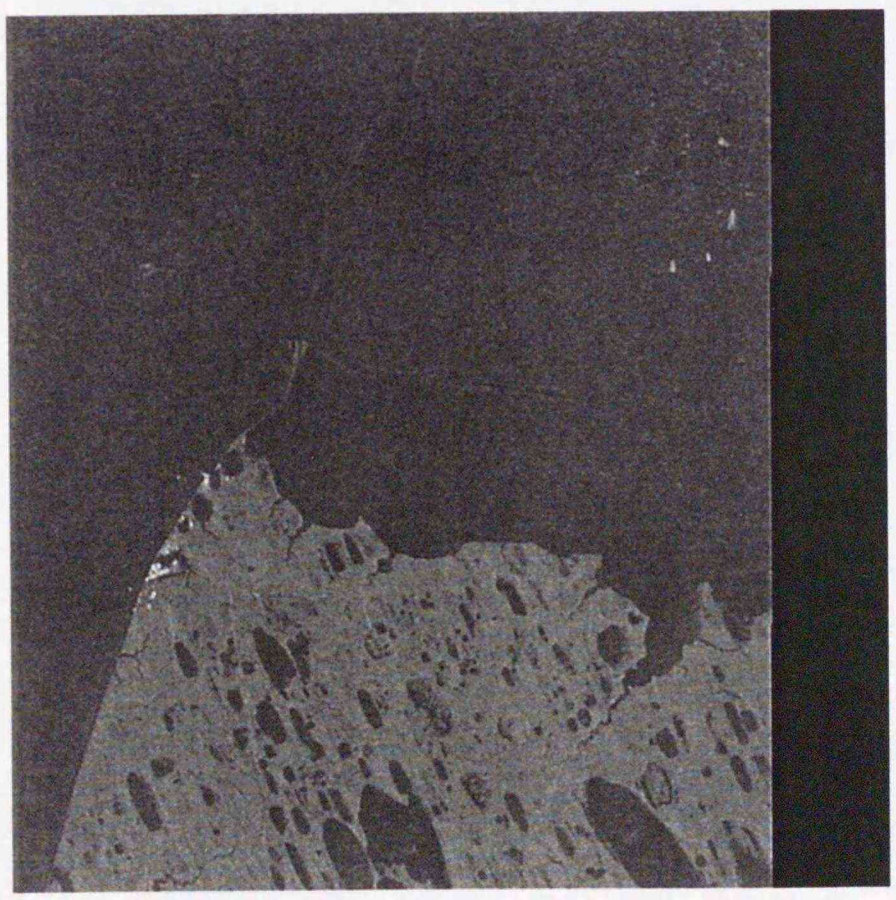
No	Image ID Path / Row	Acquisition Date	Julian Day	Ascending/ Descending
1	645 / 180	7/12/92	194/92	D
2	645 / 180	8/25/92	238/92	D
3	645 / 180	10/08/92	282/92	D
4	645 / 180	11/22/92	326/92	D
5	645 / 180	1/04/93	4/93	D
6	645 / 180	2/18/93	49/93	D
7	645 / 180	4/02/93	92/93	D

Because the EOC's standard products have full resolution (12.5 m pixel size, 20 m resolution), the speckle noise on the images is apparent. The low resolution images

(100 m pixel size) were generated by block averaging of 8 x 8 pixels in order to reduce the speckle noise, then each pixel value was converted to σ^0 related value according to the calibration equation given by the EOC calibration team [*Shimada*, 1995]. The target area near Barrow was extracted from the first image taken on July 12, 1992. The other 6 images were overlaid on the reference image coordinate using more than 8 control points. The processed images are shown in Fig. 3-2.



(1) ID: Path 645 / Row 180 (7/12/92)

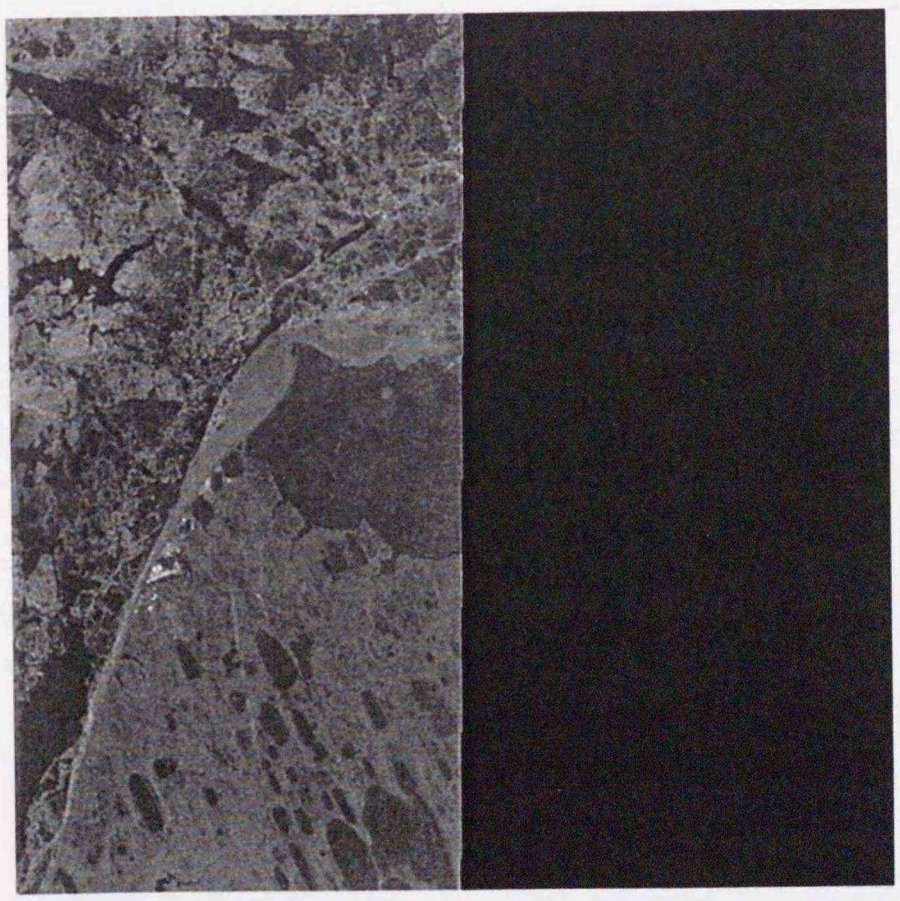


(2) ID: Path 645 / Row 180 (8/25/92)

Fig. 3-2 Time series of L-Band radar images for lakes near Barrow(1/4).

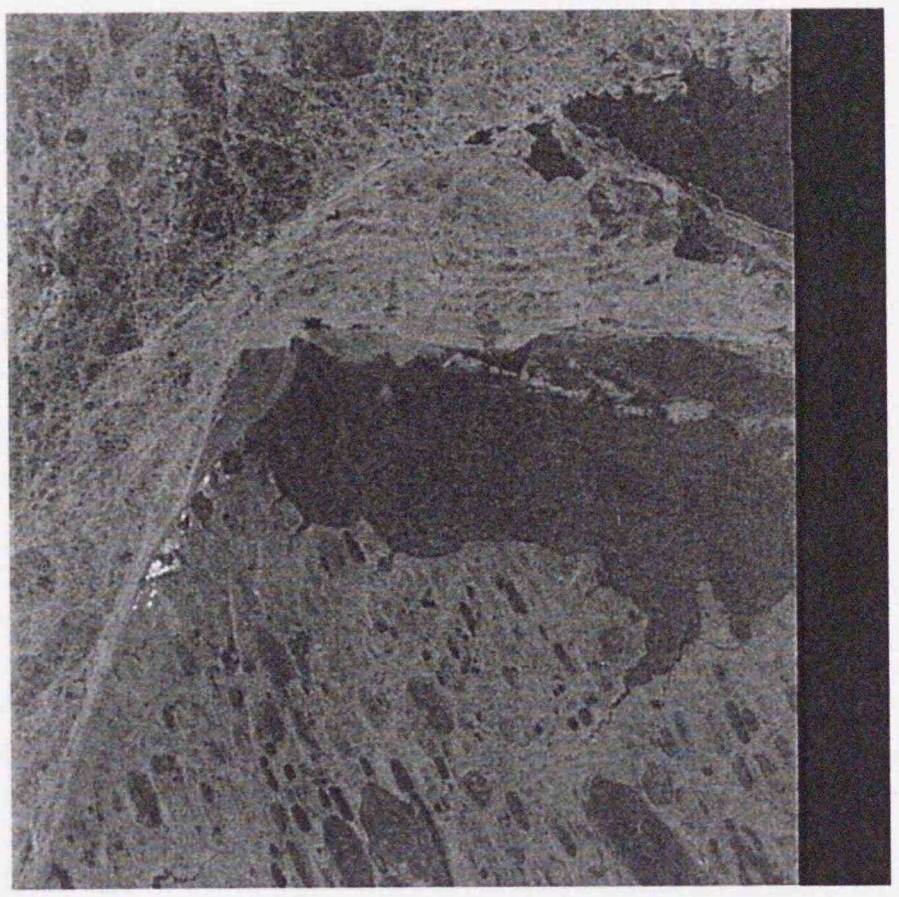


(3) ID: Path 645 / Row 180 (10/08/92)

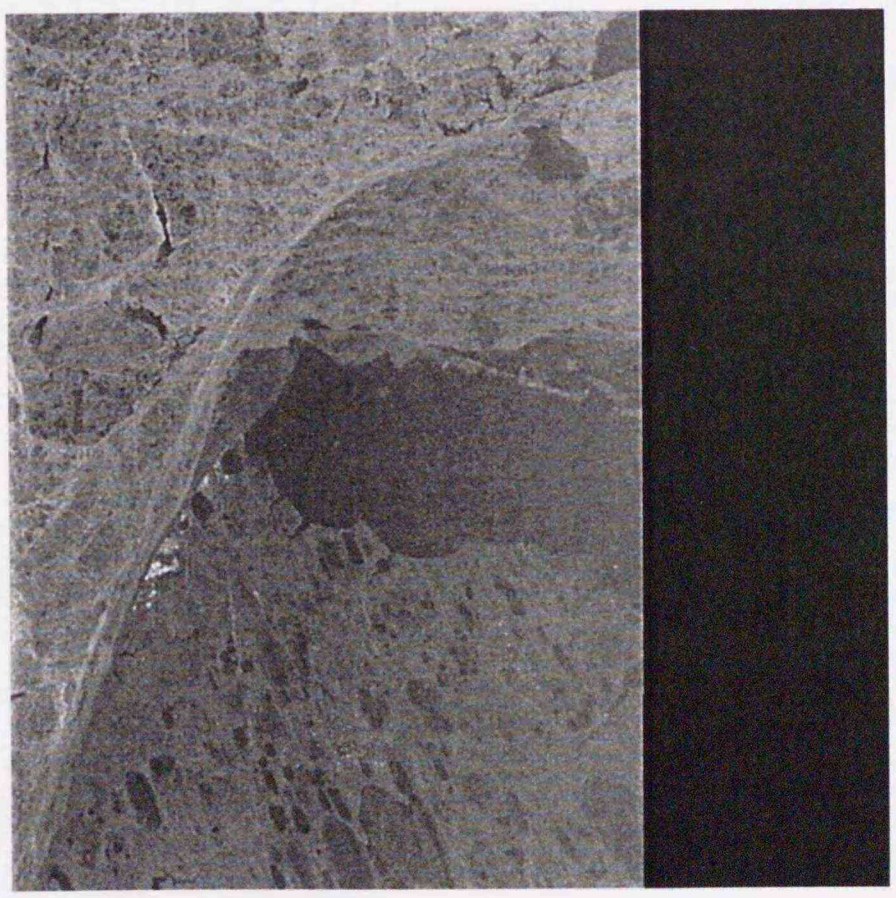


(4) ID: Path 645 / Row 180 (11/22/92)

Fig. 3-2 Time series of L-Band radar images for lakes near Barrow(2/4).



(5) ID: Path 645 / Row 180 (1/04/93)



(6) ID: Path 645 / Row 180 (2/18/93)

Fig. 3-2 Time series of L-Band radar images for lakes near Barrow(3/4).



(7) ID: Path 645 / Row 180 (4/02/93)

Fig. 3-2 Time series of L-Band radar images for lakes near Barrow(4/4).

3-2. TEMPORAL CHANGES IN BACKSCATTER

3-2.1. TIME CHANGES FOR FIELD OBSERVATION SITES IN C-BAND

To understand the time-changes of backscatter for lakes near Barrow, the backscattering coefficient for each site was extracted from SAR data. The coordinates of the field observation points on the registered image are given in Table 3-3. The changes in backscattering coefficient as a function of time are shown in Fig. 3-3.

In late September when the ice began to grow on the lakes, σ° values were in the range –21.8 to –16.0 dB. By mid to late October, values at all sites had increased to –16.0 to –11.8 dB.

From late October onward, σ° values at site 7 decreased further before leveling off in late December at -18 dB, where they remained throughout the winter. At sites 3,4 and 11, maximum σ° value of –12.6 to -11.4 dB was reached in early December, then decreased during December and remained constant at -16.5 to -14.2 dB from early January to late April.

The σ° at site 6 reached a maximum of -8.0 dB in early January, but by early February had decreased to -15.8 dB where it remained quite constant through late April. At sites 1, 2, 5, 8, 9, and 10, σ° values had increased to -8.4 to –5.4 dB by late January and remained quite constant at those values for the rest of the winter until the spring thaw period (mid May).

Table 3-3 The coordinates for sampling points on registered C-band image for each field observation point.

No.	Start pixel / line	Size of averaging area
1	220 / 143	4x4
2	223 / 150	4x4
3	241 / 159	2x2
4	239 / 168	2x1
5	176 / 139	3x3
6	142 / 203	2x2
7	196 / 145	2x2
8	251 / 052	3x3
9	248 / 058	3x3
10	177 / 124	3x3
11	328 / 197	3x3

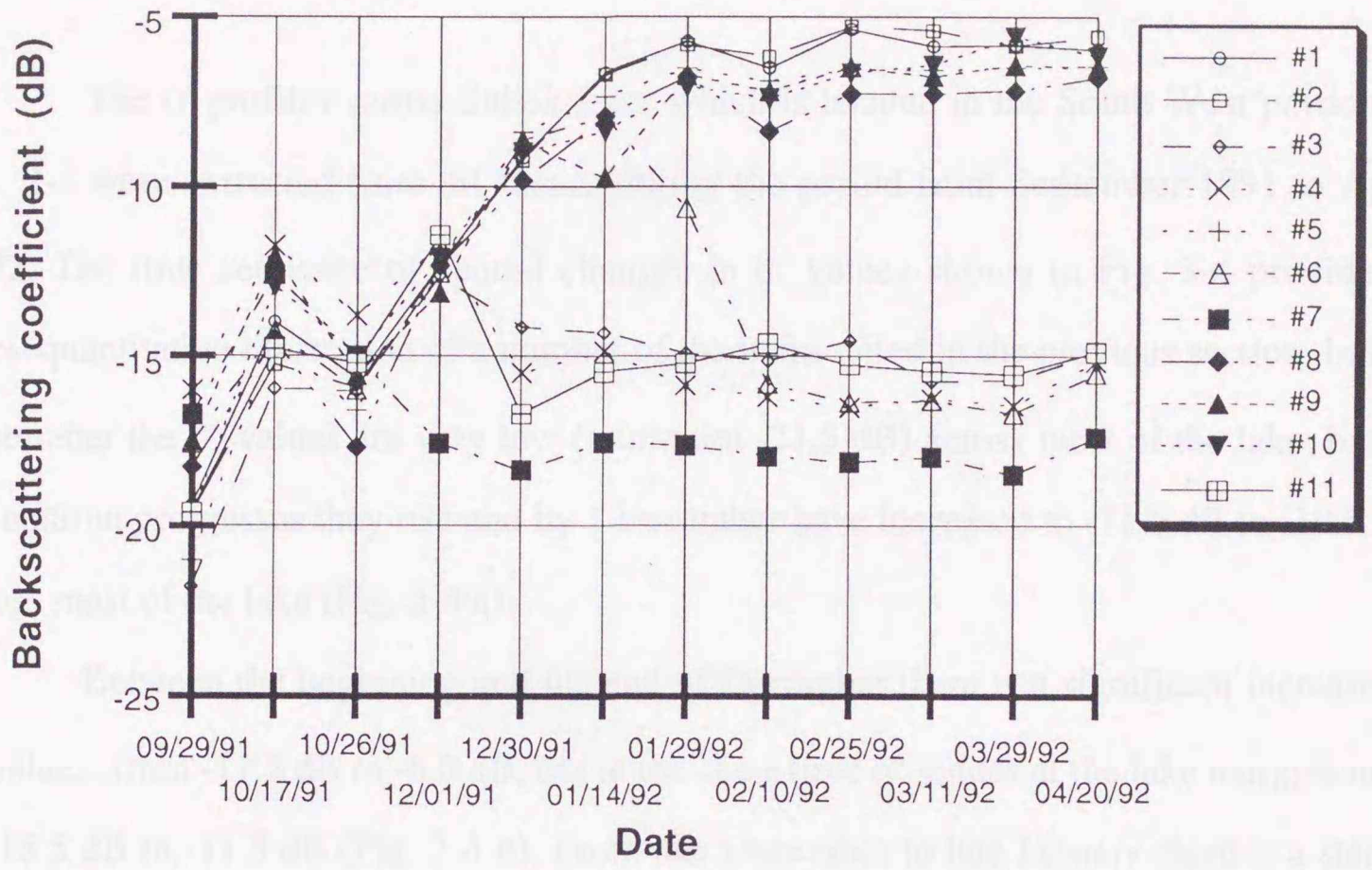


Fig. 3-3 C-band backscatter changes as a function of time for a selection of lakes during the period from 29 September 1991 to 20 April 1992.

3-2.2. SPATIAL AND TEMPORAL BACKSCATTER CHANGE AT LAKE SUKOK IN C-BAND

The σ^{o} profiles across Sukok Lake which is located in the South West portion of Fig. 2-1 were extracted from all scenes during the period from September 1991 to April 1992. The time sequence of spatial changes in σ^{o} values shown in Fig. 3-4 provides a semi–quantitative illustration of a number of the items noted in the previous section. In late September the σ^{o} values are very low (minimum -21.5 dB) across most of the lake, but as the autumn progresses they rise and by 1 December have increased to -11.5 dB to -10.5 dB across most of the lake (Fig. 3-4 a).

Between the beginning and the end of December there is a significant increase in σ^{o} values, from -11.5 dB to -6.0 dB, but at the same time σ^{o} values at the lake margins drop to -15.5 dB to -11.5 dB (Fig. 3-4 b). From late December to late January there is a steady increase in σ^{o} values to as much as -5.5 dB to -4.5 dB across the lake, with a strong discontinuity at the margins where the σ^{o} values remain at -15.5 dB to -11.5 dB (Fig. 3-4 b).

From the end of January to late April there are significant changes in the σ^{o} values (Fig. 3-4 c, d). The marginal area of low σ^{o} values increases as the σ^{o} values fall to as low as -17.5 dB, while the central area of high σ^{o} values shrinks as the σ^{o} values decrease to -8.0 dB.

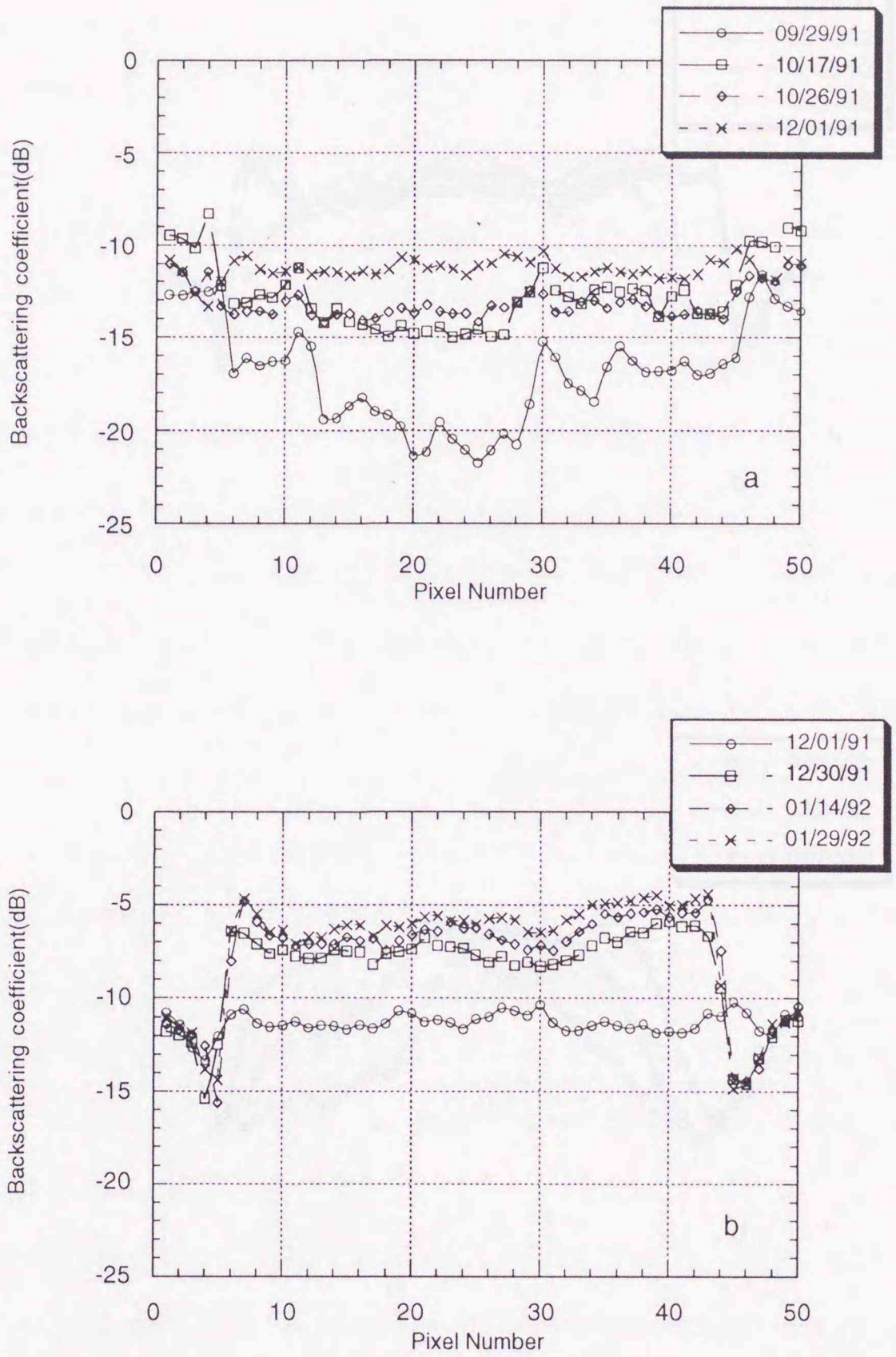
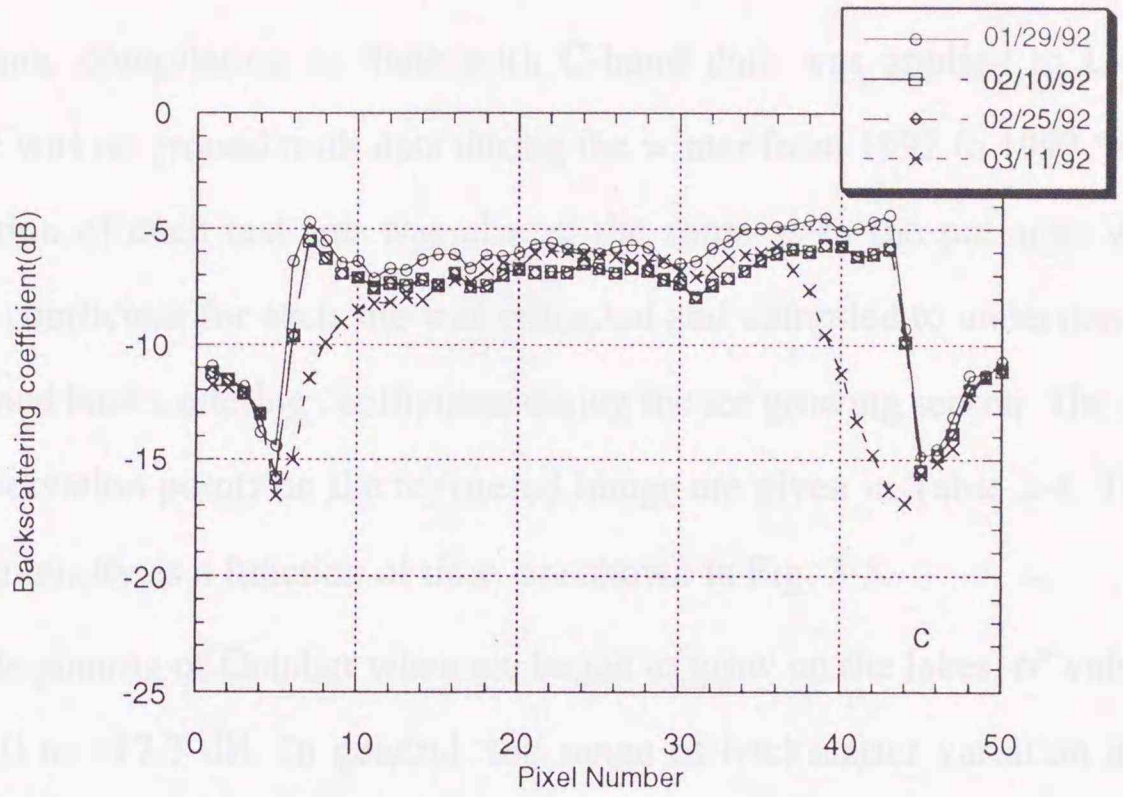


Fig. 3-4 Profiles of spatial changes in digital number σ^{o} values along the same transact across Sukok Lake from 29 September 1991 to 20 April 1992 (1/2).



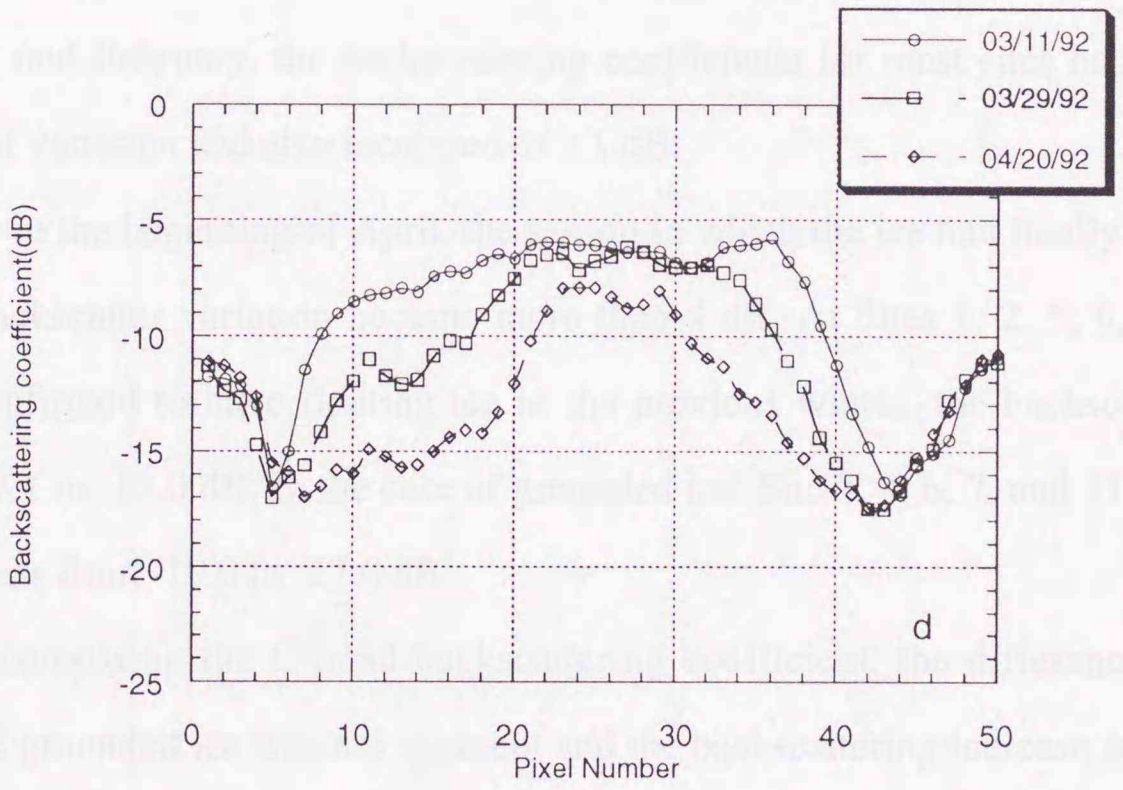


Fig. 3-4 Profiles of spatial changes in digital number σ^{o} values along the same transact across Sukok Lake from 29 September 1991 to 20 April 1992 (2/2).

3-2.3. TIME CHANGES FOR FIELD OBSERVATION SITES IN L-BAND

The same compilation as done with C-band data was applied to L-band data. Although there was no ground truth data during the winter from 1992 to 1993, we assumed that the condition of each test site was almost the same as in the previous winter. The backscattering coefficient for each site was extracted and compiled to understand the time-changes of L-band backscattering coefficients during the ice growing season. The coordinates of the field observation points on the registered image are given in Table 3-4. The changes in backscatter intensity as a function of time are shown in Fig. 3-5.

In the beginning of October when ice began to grow on the lakes, σ^{o} values were in the range –19.0 to –17.7 dB. In general, the range of backscatter variation around this season is 1.3 dB and it is smaller than the variation in C-band, which is about 5 to 6 dB.

By mid November, values at most sites had slightly increased from –18.1 to –17.3 dB. Then until mid February, the backscattering coefficients for most sites had increased and the range of variation had also increased to 2.1 dB.

Finally in the beginning of April, the season in which the ice had finally grown up, the range of backscatter variation became more than 4 dB. At Sites 1, 2, 5, 8, 9 and 10, which were confirmed to have floating ice in the previous winter, the backscatter value varied from -17.1 to -15.0 dB. In the case of grounded ice, Site 3, 4, 6, 7, and 11, the range of coefficient was from -19.0 to -17.0 dB.

When comparing the C-band backscattering coefficient, the difference between floating ice and grounded ice was not apparent and the backscattering increase for growing ice was 4 dB and much smaller than that in C-band.

Table 3-4 The coordinates for sampling points on registered L-band image for each field observation point.

No.	Start pixel / line	Size of averaging area
1	145 / 363	4x4
2	152 / 369	4x4
3	170 / 365	2x2
4	172 / 373	2x1
5	104 / 382	3x3
6	115 / 456	2x2
7	126 / 376	2x2
8	120 / 275	3x3
9	120 / 270	3x3
10	099 / 369	3x3
11	265 / 349	3x3

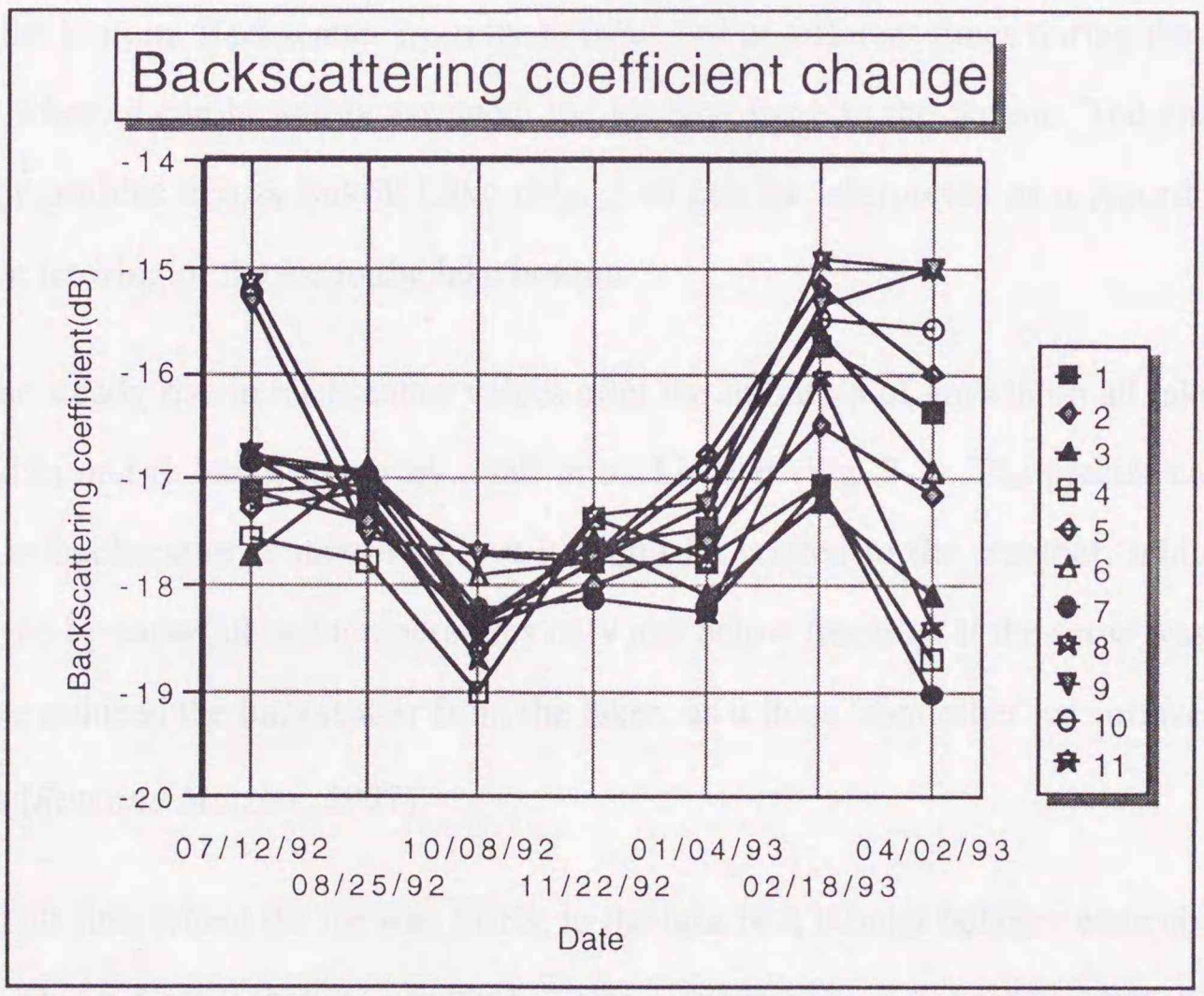


Fig. 3-5 L-band backscatter changes as a function of time for a selection of lakes.

3-3. ICE GROWTH, STRUCTURE AND BACKSCATTER VARIATIONS

3-3. 1. C-BAND BACKSCATTER

At each site that was sampled in April 1992, ice that was frozen to the lake bed corresponded to areas of low backscatter in the C-band SAR scenes, and ice that was floating corresponded to areas of high backscatter. These observations add to the evidence that the low backscatter is due to the low dielectric contrast of the ice–soil interface, resulting in strong attenuation of the radar signal and minimal returns to the radar antenna [*Sellmann et al.*, 1975; *Elachi et al.*, 1976; *Weeks et al.*, 1977, 1978, 1981; *Mellor,* 1982].

Backscatter from the ice was also low at the beginning of the growth season (Figs. 3-3 and 3-4). This can be attributed to specular reflection away from the radar antenna off the smooth horizontal surfaces (upper and lower) of the thin ice. As the ice thickened, backscatter values rose steadily, except for those lakes that were subsequently found to be frozen to the bottom. Backscatter from those lakes fell at different times during the winter (Fig. 3-3) when, it can be safely assumed, the ice first froze to the bottom. The changing backscatter profiles across Sukok Lake (Fig. 3-4) can be interpreted as a record of the progressive freezing of the ice to the lake bottom.

The steady rise in backscatter values after the initiation of growth on all lakes was interrupted in mid to late October by a fall in backscatter (Fig. 3-3). The precise cause of this drop in backscatter is unknown, but it might be related to the weather, which was characterized by snowfall and temperatures only just below freezing. If the snow was wet it would have reduced the backscatter from the lakes, as it does from other ice surfaces such as glaciers [*Rott and Mätzler*, 1987].

At all sites where the ice was frozen to the lake bed, tubular bubbles were observed in the ice. This indicates that the tubular bubbles serve largely as forward scatterers as the radar signal passes through to the base of the ice. Since there is a minimal return to the radar once the ice is frozen to the bottom, the strong backscatter that is observed in the ice

prior to freezing to the bottom (grounding) and in the ice that remains afloat all winter must require the presence of both an ice–water interface and the tubular bubbles.

An ice–water interface has a strong dielectric contrast that will give rise to specular reflection away from the radar [*Weeks et al.*, 1978]. However, at the time the lakes were sampled, there was strong backscatter at those sites where the ice was still floating. The only explanation for the strong backscatter is the presence of the tubular bubbles, which would cause volume scattering of the radar signal once it had been reflected off the ice–water interface [*Weeks et al.*, 1978].

The significant role of the tubular bubbles is illustrated by the backscatter differences between the ice that remained afloat and that which was frozen to the lake bed at the end of April. With σ° values of –6 to -5 dB, backscatter from the floating ice is significantly greater than the backscatter from the grounded ice (–18 to –15 dB). A similar large difference exists between the backscatter from ice that remained afloat in April and backscatter from the same ice at the beginning of the growth season (Sites 1 and 2, Fig. 3-3). It is also worth noting that the backscatter from the grounded ice is similar to the backscatter from the new ice that began to grow in September (Fig. 3-3). This similarity in backscatter provides an indication of the effectiveness of the new ice as a specular reflector prior to tubular bubble development.

Earlier studies of the SAR signatures of ice on shallow Arctic lakes have emphasized the importance of the tubular bubbles [*Sellmann et al.*, 1975; *Elachi et al.*, 1976; *Weeks et al.*, 1977, 1978, 1981; *Mellor*, 1982]. No previous mention has been made of the possible impact on backscatter variability of the bubbly snow–ice layer at the surface (Fig. 2-2). Some indication of the relative importance of the bubbly surface layer can be gained from a comparison of backscatter at Sites 1 and 2. Both sites were on the same lake (Fig. 2-1), but only Site 2 had a bubbly surface layer. Despite the physical difference between each site, there is a negligible difference in backscatter at each site. Thus, it appears that the effects of

the tubular bubbles far exceed any effects due to the presence of bubbly snow–ice layers.

3-3. 2. L-BAND BACKSCATTER

Although there was no ground truth data to be compared with L-band SAR data, we considered that the condition of ice at each test site was almost the same as that in the previous year, and compiled a time variation of backscattering coefficient for each test site.

In the beginning of October 1992, the backscattering coefficients for all sites were low because of specular reflection from thin ice on the lake (Fig. 3-5). From October 1992 to February 1993, the backscattering coefficient had increased 2 to 3 dB at most of the sites (Fig. 3-5). It was considered that this phenomenon was caused by the increase in thickness of the tubular bubbles layer as the ice grew. The increase of backscatter as the ice grew was very small compared with the backscatter increase of that in the C-band.

Assuming that the ice condition was the same as that in the previous year, April 1992, we divided the test sites into two groups which have floating ice (Sites 1, 2, 5, 8, 9 and, 10) and have grounded ice (Sites 3, 4, 6, 7 and, 11). The backscatter difference between the two groups in April 1993 was approximately 2 dB and much smaller than that in the C-band. The smaller difference of backscatter between the two groups is probably because a noise equivalent backscattering coefficient ($NE\sigma^0$) of JERS-1 SAR is relatively high. The specification of JERS-1's $NE\sigma^0$ shows -20.5 dB, however at this time the SAR is operating on reduced transmitter power because of the failure of its equipments.

CHAPTER 4. LAKE ICE BACKSCATTER MODEL

Based on the field observation and ice core analysis described in chapter 2, a multi-layer backscatter model comprised of the following elements was developed: (1) specular air-ice boundary; (2) specular or slightly rough ice-water and ice-frozen soil boundaries; (3) an ice layer of variable thickness; (4) ice sub-layers with air inclusions of variable density, size and shape, including spheres, prolate spheroids, and cylinders of finite length.

This chapter reports on the development of the numerical model of microwave backscatter from ice covered, shallow tundra lakes.

4-1. LAKE ICE FEATURES CONSIDERED IN MODELING

When considering microwave scattering models for sea ice, the scattering from the ice surface is dominant scattering in most cases [*Wakabayashi et al.* 1994b, 1995b]. When comparing scattering from lake ice with that from sea ice, because the dielectric loss factor for lake ice is generally very low, we should consider an ice volume scattering as well as an ice bottom scattering .

The result of field observations and ice core measurements indicated the presence of three ice types: (1) granular ice containing roughly spherical bubbles, with radii much smaller than a wavelength ; (2) clear bubble-free ice; and, (3) ice containing tubular bubbles resembling thin cylinders. The latter, oriented vertically to the ice surface, also had radii much smaller than a wavelength, but with lengths ranging from 1.5 cm to 9.1 cm. The number of tubular bubbles per square meter, i.e. bubble density or ice porosity, was quite variable [*Jeffries et al.*, 1993, 1994; *Wakabayashi et al.*, 1992].

The typical stratigraphy of the lake ice is shown in Fig. 4-1. The clear ice layer and one or more tubular bubble layers were found at all sites, while the granular ice layers were found at 6 of the 10 sites. Each layer had a different thickness, bubble density

and bubble size, especially in the tubular bubble layer where the bubble density and bubble size varied considerably with depth. The snow cover on the ice had variable depth (0-80 mm) and was dry.

Considering the features described above, the basic lake ice model is comprised of the following elements [*Wakabayashi et al.*, 1993] :

(1) three main layers (either air-ice-water or air-ice-frozen soil);

(2) sub-layers in the ice layer with each sub-layer having a different bubble density and bubble size (a bubble density of zero corresponds to clear ice);

(3) the upper boundary of the ice is considered to be specular and the bottom boundary to be specular or slightly rough;

(4) reflections occur at each boundary and at the air bubbles; and,

(5) dielectric parameters of the ice layers change with the bulk ice density, i.e., with the density of bubbles present in the ice.

Since the surface roughness of the ice cover on these lakes was confirmed to be small by field observations, a surface scattering from the ice can be neglected. The dry snow cover can also be neglected by the following reasons. Since for a dry snow the dielectric mismatch at the air/snow boundary is very small, the reflection at the air-snow boundary can be ignored. In addition, since the snow cover was thin (less than 8 cm for all sites in the field observation) and snow particles were very small, the contribution of the snow cover to the backscattering coefficient may be neglected at both L- and C-band frequencies [*Ulaby et al.*, 1986]. It should be noted that reflections are added incoherently in the model, because the length of each path is widely distributed. Calculations are made on the basis of power, because each scattering has a different scattering path.

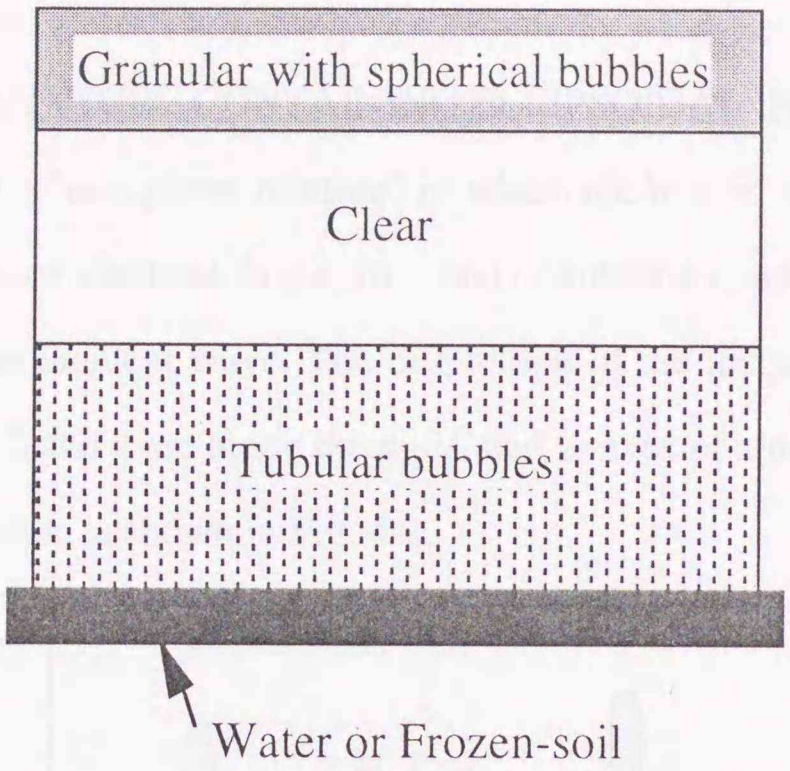


Fig. 4-1 Typical structure of lake ice as observed in Northern Alaska.

4-2. AVERAGE DIELECTRIC CONSTANT FOR ICE

The lake ice on the North Slope is taken to be a mixture consisting only of pure ice and air bubbles. Individual bubbles in the tubular bubbles layer are assumed to be strongly elongated in the z-direction which is perpendicular to the air/ice interface. The bubble shapes are approximated as spheres in the granular layer and narrow cylinders in the tubular bubbles layer. Under these assumptions, the average dielectric constant for the ice and air mixture can be obtained.

The average dielectric constant of a heterogeneous mixture consisting of two or more than two substances is related to the dielectric constants of the individual substances, their volume fractions, their spatial distributions, and their orientations relative to the direction of the incident electric-field vector. Usually the substance with the highest volume fraction is regarded as the "host" material and the other substance or substances are regarded as "inclusions". Several comprehensive studies of dielectric mixing models have been published in available literature [*de Loor*, 1968]. Most of the available theoretical models have been developed for media that contain only particles of some special shape such as spheres, ellipsoids, needles, and disks .

4-2.1. TWO-PHASE MIXTURE CONTAINING ELLIPSOIDAL INCLUSIONS

First consider a "two-phase mixture" in which the host is ice and the inclusions are ellipsoidal particles of identical shape, size, and orientation relative to the direction of the electric field of the incident wave. The orientation of the inclusions is chosen such that their semi-axes a, b and c are along the x-, y- and z- axes of a rectangular coordinate system. This configuration is shown in Fig. 4-2.

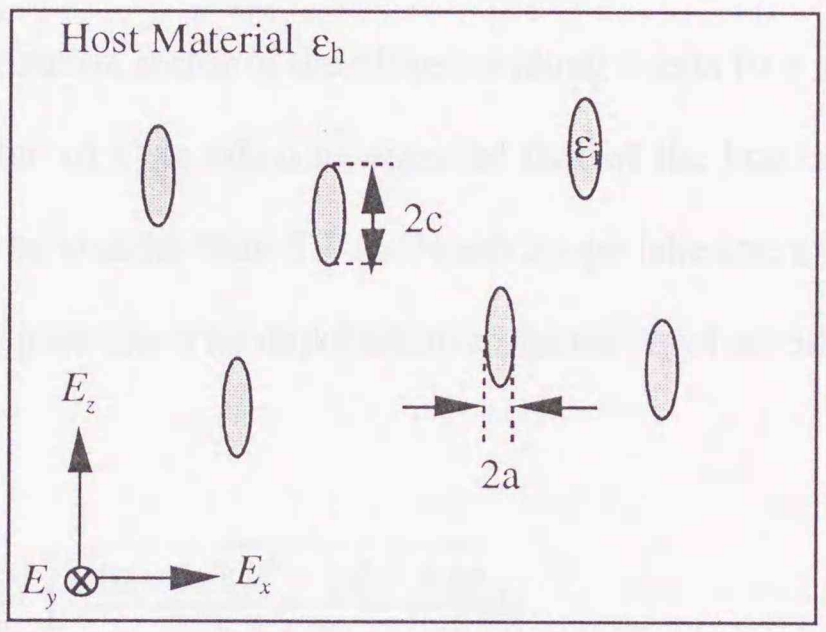


Fig. 4-2 Configuration of two-phase mixture model.

The volume inclusion factor v_i is defined as

$$v_i = \frac{4}{3}\pi abcN \ , \tag{4.1}$$

where N is the number of ellipsoids per unit volume of the mixture. The volume fraction of the host material v_h is

$$v_h = 1 - v_i \ . \tag{4.2}$$

Although the dielectric constants for host and inclusions are taken to be isotropic, the average dielectric constant of the mixture is generally anisotropic because the polarizability of the ellipsoidal particles are governed by the direction of the applied electric field E relative to the orientation of the ellipsoid. Therefore the dielectric constants of the mixture model should be calculated in the x, y and z directions, and then be

derived in the desired direction. De Loor [1968] derived the following expression for the average dielectric constant of a two-phase mixture ε_m:

$$\varepsilon_m = \varepsilon_h + \frac{v_i(\varepsilon_i - \varepsilon_h)}{1 + A_u(\frac{\varepsilon_i}{\varepsilon^*} - 1)} , \tag{4.3}$$

where ε_h and ε_i are the relative dielectric constants for the host and inclusion respectively, ε^* is an effective dielectric constant for the region immediately surrounding an included particle, A_u is the depolarization factor of the ellipsoid along u-axis (u = a, b or c). If v_i is smaller than 0.1, the value of ε^* is taken as equal to that of the host material. As this factor usually appears to be smaller than 0.1 for North Slope lake ice, ε^* is taken to have the same value of that of pure ice. The depolarization factor A_u of an ellipsoid along the u-axis is given by

$$A_u = \frac{abc}{2}\int_0^\infty \frac{\sqrt{(a^2+s)(b^2+s)(c^2+s)}}{(u^2+s)} ds , \tag{4.4}$$

where u = a, b or c, and

$$A_a + A_b + A_c = 1 . \tag{4.5}$$

4-2.2. MIXTURE MODEL FOR LAKE ICE

The lake ice in North Slope area can be considered as a special case of a two phase mixture with the volume inclusion less than 10 %. The host material is considered as pure ice containing inclusions of air in the shape of spheres for the granular ice, or long and narrow cylinders for the tubular bubbles layer.

In case of the granular layer, the average dielectric constant is given by

$$A_a = A_b = A_c = \frac{1}{3},\ \varepsilon_m^x = \varepsilon_m^y = \varepsilon_m^z = \varepsilon_m \text{ , and}$$

$$\varepsilon_m = \varepsilon_h\left[1 + 3v_i\frac{\varepsilon_i - \varepsilon_h}{\varepsilon_i + 2\varepsilon_h}\right] . \tag{4.6}$$

For the tubular bubbles layer, the bubbles are oriented along z-axis and dispersed in the host material, equation (4.3) becomes:

$$A_a = A_b = 0.5,\ A_c = 0\ ,$$

$$\varepsilon_m^x = \varepsilon_m^y = \varepsilon_h\left[1 + 2v_i\frac{\varepsilon_i - \varepsilon_h}{\varepsilon_i + \varepsilon_h}\right], \text{ and}$$

$$\varepsilon_m^z = \varepsilon_h + v_i(\varepsilon_i - \varepsilon_h)\ . \tag{4.7}$$

4-3. REFLECTIVITY OF BOUNDARY

As the snow covers on North Slope lakes are thin, of a low density, and (during the winter) dry, they will be neglected. The ice surface is considered to be flat. In addition, ice thickness on a given lake is considered to be constant which implies that the ice and water boundary is also flat. To make the initial lake ice model as simple as possible, both the air-ice and ice-water / ice-soil boundaries will be considered as specular surfaces. Also the boundaries between ice sub-layers can be treated as a specular surface with small error, because the dielectric mismatch at the sub-layer boundary is considered to be very small. According to the calculation of an average dielectric constant based on two-phase mixture model, the power reflectivity at the ice sub-layers boundary is much smaller than 0.001 at incidence angle of 0-40 deg., even if the difference of air inclusion factor is 10 % between two sub-layers.

For these assumptions Fresnel's law gives the reflection and transmission at the boundaries for H- and V-polarization as follows:

For H-polarization

$$R_H = \frac{\eta_2\cos\theta_1 - \eta_1\cos\theta_2}{\eta_2\cos\theta_1 + \eta_1\cos\theta_2} = \frac{\frac{\mu_2}{\mu_1}\cos\theta_1 - \sqrt{\frac{\mu_2\varepsilon_2}{\mu_1\varepsilon_1} - \sin^2\theta_1}}{\frac{\mu_2}{\mu_1}\cos\theta_1 + \sqrt{\frac{\mu_2\varepsilon_2}{\mu_1\varepsilon_1} - \sin^2\theta_1}}\ , \tag{4.8}$$

$$T_H = \frac{2\eta_2 \cos\theta_1}{\eta_2 \cos\theta_1 + \eta_1 \cos\theta_2} , \tag{4.9}$$

For V-polarization

$$R_V = \frac{\eta_1 \cos\theta_1 - \eta_2 \cos\theta_2}{\eta_1 \cos\theta_1 + \eta_2 \cos\theta_2} = \frac{\frac{\varepsilon_2}{\varepsilon_1}\cos\theta_1 - \sqrt{\frac{\mu_2\varepsilon_2}{\mu_1\varepsilon_1} - \sin^2\theta_1}}{\frac{\varepsilon_2}{\varepsilon_1}\cos\theta_1 + \sqrt{\frac{\mu_2\varepsilon_2}{\mu_1\varepsilon_1} - \sin^2\theta_1}} , \tag{4.10}$$

$$T_V = \frac{2\eta_1 \cos\theta_1}{\eta_1 \cos\theta_1 + \eta_2 \cos\theta_2} . \tag{4.11}$$

Each term is described in Fig. 4-3.

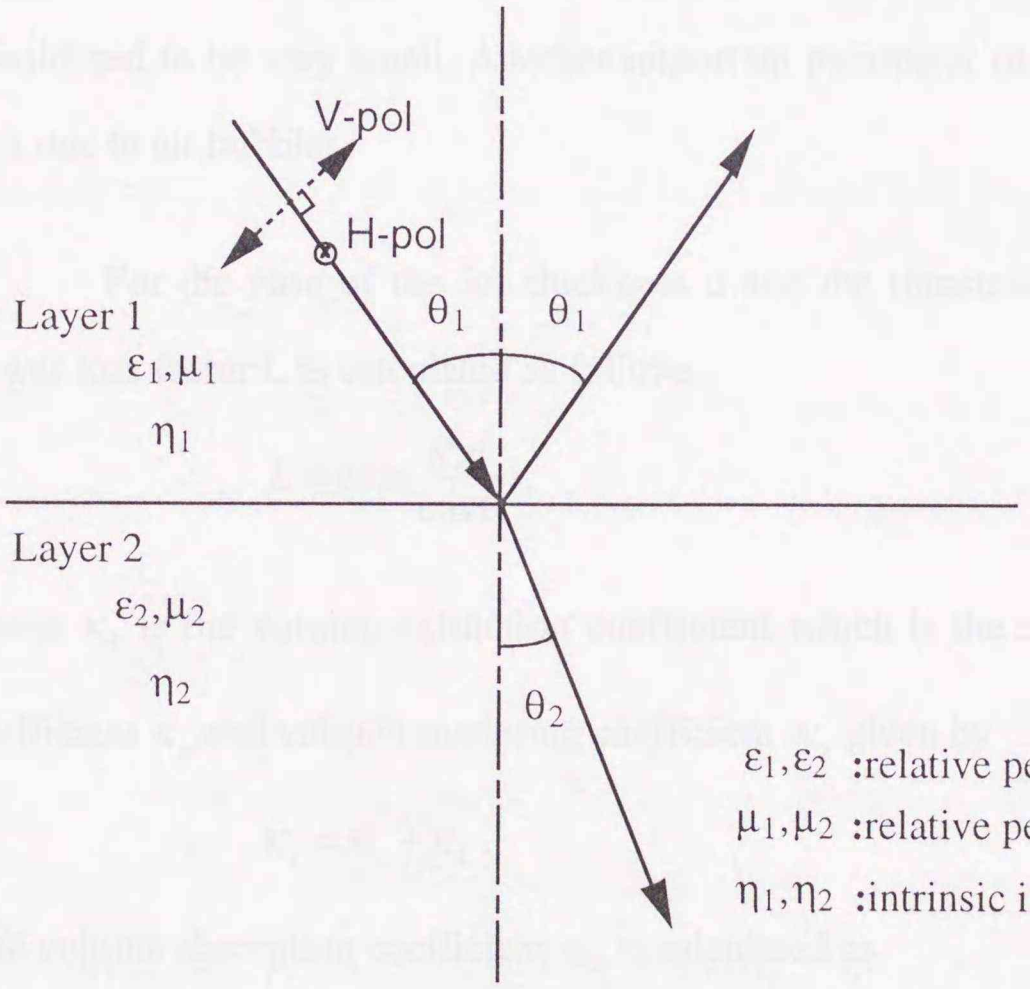


Fig. 4-3 Reflection and transmission at a specular boundary.

R_H, T_H and R_V, T_V are the reflection and transmission coefficients for H- and V-polarization when the incidence angle is θ_1. In order to make a model for lake ice, the power reflectivity (Γ_H, Γ_V) and power transmissivity (γ_H,γ_V) values are also required.

These are calculated from equations (4.12) - (4.15):

$$\Gamma_H = |R_H|^2, \tag{4.12}$$

$$\Gamma_V = |R_V|^2, \tag{4.13}$$

$$\gamma_H = 1 - |R_H|^2, \text{ and} \tag{4.14}$$

$$\gamma_V = 1 - |R_V|^2 . \tag{4.15}$$

4-4. POWER LOSS FACTOR OF ICE

Fresh-water ice is only a slightly lossy material in that the imaginary part of the dielectric constant is very small for the frequencies in L- and C- band. By comparison, the imaginary part of the dielectric constant for first-year sea ice is more than 100 times larger than that of fresh water ice. Therefore an absorption loss of freshwater ice is considered to be very small. Another important parameter for loss factor is a scattering loss due to air bubbles.

For the case of the ice thickness d and the transmission angle θ, an one-way power loss factor L is calculated as follows:

$$L = \exp\left(\frac{\kappa_e d}{\cos\theta}\right), \tag{4.16}$$

where κ_e is the volume extinction coefficient which is the sum of volume absorption coefficient κ_a and volume scattering coefficient κ_s given by

$$\kappa_e = \kappa_a + \kappa_s . \tag{4.17}$$

The volume absorption coefficient κ_a is calculated as

$$\kappa_a = 2k_0\left[\frac{1}{2}\left(-\varepsilon' + \sqrt{(\varepsilon')^2 + (\varepsilon'')^2}\ \right)\right]^{\frac{1}{2}}, \tag{4.18}$$

where k_0 is the wave number in free space, ε' and ε'' are real and imaginary parts of the average dielectric constant.

Considering the lake ice as a homogeneous medium (fresh water ice) randomly embedded with small particles (air bubbles), each particle is characterized by a bistatic radar cross section σ_p due to a p-polarized (p = H or V) incident wave. The scattering cross-section of the particle Q_{sp} is defined as the cross section which produces the total scattered power surrounding the particle due to a unit incident Poyinting vector of polarization p:

$$Q_{sp}(\theta,\phi) = \frac{1}{4\pi}\int_{4\pi} \sigma_p d\Omega_s \ , \tag{4.19}$$

where θ and ϕ are the direction of incident wave and this integration is done over the scattered solid angle. The volume scattering coefficient for the inhomogeneous medium and polarization is calculated as

$$\kappa_s = Q_{sp} \times N \ , \tag{4.20}$$

where N is the number of particles within unit volume [*Ishimaru*, 1978].

As shown above, the power loss factor L is calculated by (4.16) - (4.20). The larger the incident angle is, the bigger is the power loss factor, because the length of the path in the ice is proportional to sec(θ).

4-5. BUBBLE SCATTERING

According to the results of ice core measurements, we have to make three types of bubble models. In the tubular bubble ice layer, the air bubbles are modeled as either prolate spheroids or cylinders of finite length according to the length of bubbles. The air bubbles in the granular ice layer are modeled as spheres with a size much smaller than a wavelength. Because the radii of the bubbles are much smaller than a wavelength for all cases, low frequency approximation is applied to obtain the bistatic cross section. The scattering formulation used in this model is that given by Ruck et al. (1970). The geometry of the problem is shown in Fig. 4-4. Let a plane wave whose propagating direction lies in

x-z plane be incident upon a bubble whose axis coincides with z axis. The direction of propagation of the incident wave makes an angle ψ_i in x-z plane. The interesting scattered field from the bubble propagates in a direction which makes an angle ψ_s in a plane rotated with the angle ϕ around the z axis.

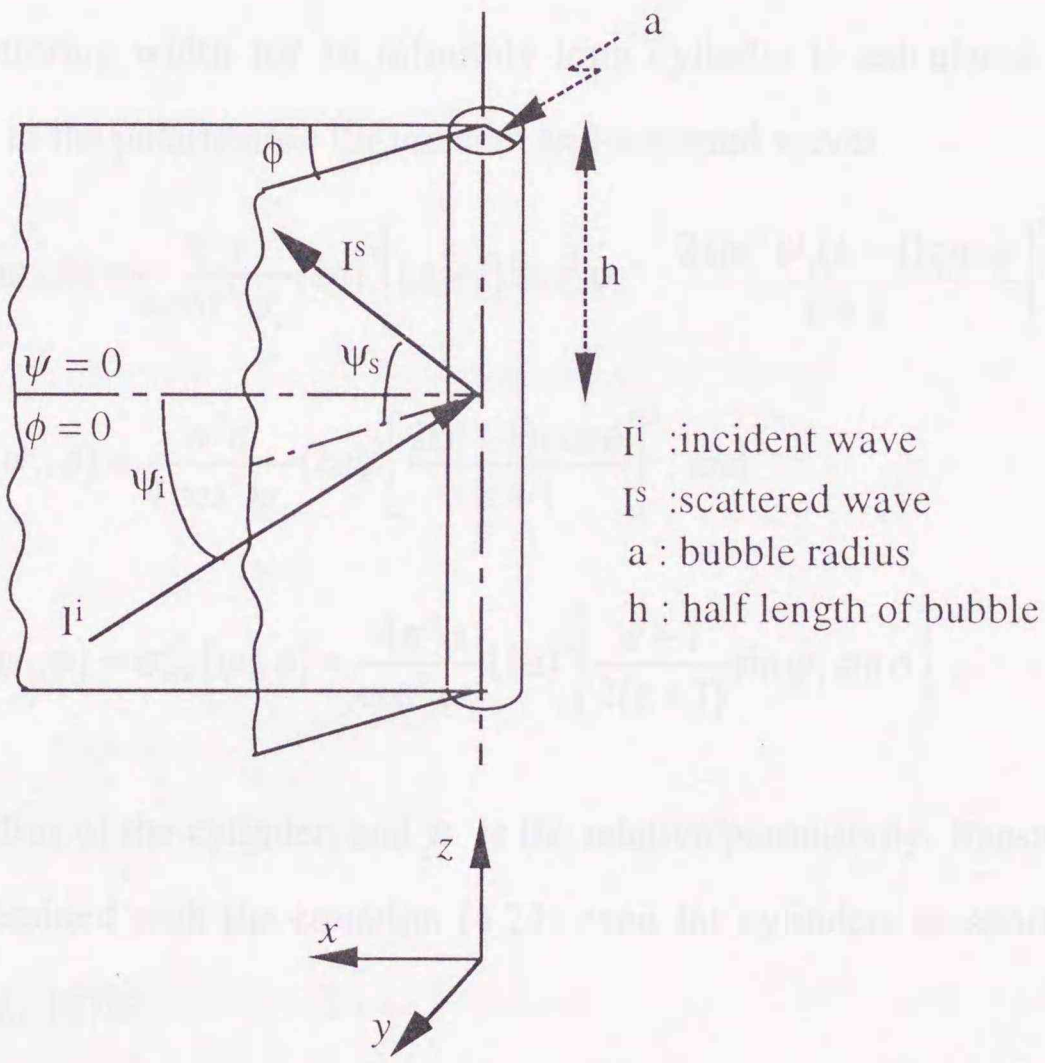


Fig. 4-4 Geometry of bubbles considered in the lake ice model.

4-5.1 TUBULAR BUBBLES WITH LENGTH GREATER THAN WAVE LENGTH

In the case that the bubble length is greater than wave length, the elongated air bubbles are modeled as identical , of finite length , vertically oriented , homogeneous dielectric cylinders. Scattering by bubbles is computed according to the formulation for a finite-length dielectric cylinder given by Ruck *et al.* (1970). The bistatic scattering

cross section of the cylinder is as follows:

$$\sigma^{cyl}(\psi_i,\psi_s,\phi)=\frac{4kh^2}{\pi}\cos^2\psi_i\sigma^c(\psi_i,\phi)\left[\frac{\sin\{k(\sin\psi_i+\sin\psi_s)h\}}{k(\sin\psi_i+\sin\psi_s)h}\right]^2, \qquad (4.21)$$

where k is the wave number , h is the half length of cylinder , $\sigma^c(\psi_i,\phi)$ is the scattering width for an infinitely long cylinder at oblique incidence.

The scattering width for an infinitely long cylinder is calculated by (4.22) - (4.24) according to the polarization for incident and scattered waves:

$$\sigma^C_{VV}(\psi_i,\phi)=\frac{\pi^2 a}{4\cos^2\psi_i}(ka)^3\left[(\varepsilon-1)\cos^2\psi_i-\frac{2\sin^2\psi_i(\varepsilon-1)\cos\phi}{\varepsilon+1}\right]^2, \qquad (4.22)$$

$$\sigma^C_{HH}(\psi_i,\phi)=\frac{\pi^2 a}{4\cos^2\psi_i}(ka)^3\left[\frac{2(\varepsilon-1)\cos\phi}{\varepsilon+1}\right]^2, \text{ and} \qquad (4.23)$$

$$\sigma^C_{VH}(\psi_i,\phi)=\sigma^C_{HV}(\psi_i,\phi)=\frac{4\pi^2 a}{\cos^2\psi_i}(ka)^3\left[\frac{\varepsilon-1}{2(\varepsilon+1)}\sin\psi_i\sin\phi\right]^2, \qquad (4.24)$$

where a is the radius of the cylinder, and ε is the relative permittivity. Reasonably good results can be obtained with the equation (4.21) even for cylinders as short as a wave length [*Ruck et al.*, 1970].

The refraction angle within the ice is given by Snell's law when the appropriate incident angle and the dielectric constants for the two layers are used. The transmitted wave first propagates downward, then is reflected at the ice-water boundary and propagates upward. If there is a bubble in the propagation path, a part of the intensity would be reflected by the bubble. Although the scattering intensity has power in all directions according to the equation (4.21), the dominant part of the scattering intensity exists in the following conditions: $\phi=0$ or π , and ψ_s is near $-\psi_i$ when the length of bubble is

much larger than the wavelength. Therefore the bubbles are regarded as cylinders which can only generate upward-going (downward-going) reflections when they are illuminated by an upward-going (downward-going) wave. Based on the above assumption, two types of scattering within the ice layer are considered. One is forward scattering and another is specular scattering. The forward scattering causes extinction of the forward propagating intensity and the specular scattering gives the backscattering intensity. The one way forward propagating loss is calculated by using the equations (4.16) - (4.20) in the previous section. The one way specular scattering reflectivity Γ_s is given by

$$\Gamma_s = \frac{Nd}{4\pi \cos\psi_i} \sigma^{cyl}(\psi_i, -\psi_i, 0) \ . \tag{4.25}$$

When the bubble length becomes shorter, a direct scattering from the bubble into the backscattering direction should be considered, in addition to the specular scattering and the forward scattering from . The one way direct scattering reflectivity $\Gamma_d(\phi)$ is calculated by setting $\psi_s = \psi_i$ and given by

$$\Gamma_d(\phi) = \frac{N\sigma^{cyl}(\psi_i, \psi_i, \phi)}{8\pi\kappa_e}\left(1 - \frac{1}{L^2}\right) . \tag{4.26}$$

4-5.2 TUBULAR BUBBLES WITH LENGTH SMALLER THAN WAVE LENGTH

When the bubble length is smaller than wavelength, all dimensions of bubbles are small compared with the wavelength. Since the target features which affect the scattering characteristics are its volume and the frequency in the low frequency region, the shape of the object is not important for calculating scattering from the object. When the bubble length is greater than its diameter, in case of the tubular bubbles layer, the bubble can be modeled by a prolate spheroid with small error [*Ruck et al.,* 1970]. When the coordinate system is taken the same as that in Fig. 4-4, the bistatic scattering cross sections for a prolate spheroid are given (4.27) - (4.30):

$$\sigma_{VV}^{elp}(\psi_i,\psi_s,\phi)=\frac{16\pi}{9}k^4\cdot$$

$$\left[(\varepsilon-1)\left\{\frac{\cos\phi\cos\psi_i\cos\psi_s}{\left((\varepsilon-1)I_a+\frac{2}{a_eb_e^2}\right)\sqrt{1-\sin^2\phi\cos^2\psi_s}}+\frac{\sin\psi_i\sin\psi_s}{\left((\varepsilon-1)I_b+\frac{2}{a_eb_e^2}\right)\sqrt{1-\sin^2\phi\cos^2\psi_s}}\right\}\right]^2, \tag{4.27}$$

$$\sigma_{HH}^{elp}(\psi_i,\psi_s,\phi)=\frac{16\pi}{9}k^4\cdot\left[(\varepsilon-1)\frac{\sqrt{1-\sin^2\phi\cos^2\psi_s}}{(\varepsilon-1)I_b+\frac{2}{a_eb_e^2}}\right]^2, \tag{4.28}$$

$$\sigma_{VH}^{elp}(\psi_i,\psi_s,\phi)=\frac{16\pi}{9}k^4\cdot$$

$$\left[(\varepsilon-1)\left\{\frac{\sin\phi\cos\psi_i\sin\psi_s\cos\psi_s}{\left((\varepsilon-1)I_a+\frac{2}{a_eb_e^2}\right)\sqrt{1-\sin^2\phi\cos^2\psi_s}}-\frac{\cos\phi\sin\phi\sin\psi_i\cos^2\psi_s}{\left((\varepsilon-1)I_b+\frac{2}{a_eb_e^2}\right)\sqrt{1-\sin^2\phi\cos^2\psi_s}}\right\}\right]^2, \tag{4.29}$$

$$\sigma_{HV}^{elp}(\psi_i,\psi_s,\phi)=0. \tag{4.30}$$

where a_e and b_e are the semi-axes of the equivalent spheroid, I_a and I_b are elliptic functions, and the other parameters are given as follows:

$$a_e=\left(\frac{3}{2}\right)^{\frac{1}{3}}h,\ b_e=\left(\frac{3}{2}\right)^{\frac{1}{3}}a,$$

$$I_a=\left(\frac{2}{a_eb_e^2}\right)\left(\frac{1}{e^2-1}\right)\left\{\frac{e}{2\left(e^2-1\right)^{\frac{1}{2}}}\ln\left(\frac{e+\left(e^2-1\right)^{\frac{1}{2}}}{e-\left(e^2-1\right)^{\frac{1}{2}}}\right)-1\right\},$$

$$I_b=\left(\frac{2}{a_eb_e^2}\right)\left(\frac{e}{2\left(e^2-1\right)}\right)\left\{e-\frac{e}{2\left(e^2-1\right)^{\frac{1}{2}}}\ln\left(\frac{e+\left(e^2-1\right)^{\frac{1}{2}}}{e-\left(e^2-1\right)^{\frac{1}{2}}}\right)\right\},$$

$$e = \frac{a_e}{b_e} \ .$$

4-5.3 SPHERE BUBBLES

In order to consider a spherical bubble model, we should set a=h for the bubble geometry in Fig. 4-4. The backscattering model for a sphere much smaller than the wavelength is also given by Ruck *et al.* (1970):

$$\sigma_{VV}^{sph}(\psi_i,\psi_s,\phi) = \sigma_{HH}^{sph}(\psi_i,\psi_s,\phi) = 4\pi a^2 (ka)^4 \left[\frac{\varepsilon - 1}{\varepsilon + 2}\right]^2 \cos^2(\psi_i - \psi_s)\cos^2\phi, \quad (4.31)$$

$$\sigma_{VH}^{sph}(\psi_i,\psi_s,\phi) = \sigma_{HV}^{sph}(\psi_i,\psi_s,\phi) = 4\pi a^2 (ka)^4 \left[\frac{\varepsilon - 1}{\varepsilon + 2}\right]^2 \sin^2\phi. \quad (4.32)$$

4-6. MULTI-LAYER CONSIDERATION

The scattering coefficient of an extended target in a given direction is the ratio of the total scattered power from an equivalent isotropic scatterer which generates the same scattered power density in the direction as the total power of the illumination area. Mathematically, the bistatic scattering coefficient is defined as

$$\sigma_{pq}^0 = \frac{4\pi r^2 \left\langle \left| E_p^s \right|^2 \right\rangle}{A \left| E_q^i \right|^2}, \quad (4.33)$$

where σ_{pq}^0 is the bistatic scattering coefficient corresponding to a q-polarized incident plane wave that gives rise to a p-polarized spherical wave, r is the distance between the center of a distributed target and the observation point, A is the illuminated area, E_q^i is the incident q-polarized field, E_p^s is the scattered p-polarized field, and $\langle\ \rangle$ is the ensemble

average.

If the incident and scattering intensities (power density) are given, the backscattering coefficient would be as follows:

$$\sigma_{pq}^{0} = 4\pi \cos\theta_0 \frac{I_p^s}{I_q^i}, \tag{4.34}$$

where θ_0 is the incident and scattered angle, I_p^s is the scattered p-polarized intensity, and I_q^i is the incident q-polarized intensity. When reflections from bubbles in the ice occur, the scattered power is propagated toward the backscattered direction, as shown schematically in Fig. 4-5. In the general case, the bubble density and bubble size vary according to the depth of ice. Thus, it is assumed that the ice is composed of n-layers and each layer has a different bubble density and bubble size. The downward and the upward intensities

$(\tilde{D}, \tilde{U})$ at the boundaries of layer i are defined as shown in Fig. 4-6. The relation of these intensities is given by

$$\begin{bmatrix} \tilde{D}_i \\ \tilde{U}_{i-1} \end{bmatrix} = \begin{bmatrix} M_a & M_b \\ M_c & M_d \end{bmatrix} \begin{bmatrix} \tilde{D}_{i-1} \\ \tilde{U}_i \end{bmatrix}, \tag{4.35}$$

where

$$\tilde{D}_i = \begin{bmatrix} D_i^+ & D_i^- \end{bmatrix}^t, \ \tilde{D}_{i-1} = \begin{bmatrix} D_{i-1}^+ & D_{i-1}^- \end{bmatrix}^t, \ \tilde{U}_i = \begin{bmatrix} U_i^+ & U_i^- \end{bmatrix}^t, \ \tilde{U}_{i-1} = \begin{bmatrix} U_{i-1}^+ & U_{i-1}^- \end{bmatrix}^t.$$

Here t indicates transpose, and M_a, M_b, M_c and M_d are 2×2 matrices, of which the elements are given by the calculation for the first order scattering. M_a, M_b, M_c and M_d are given as follows:

$$M_a = \begin{bmatrix} a_1 & a_2 \\ a_2 & a_1 \end{bmatrix}, \ M_b = \begin{bmatrix} b_1 & b_2 \\ b_2 & b_1 \end{bmatrix}, \ M_c = \begin{bmatrix} c_1 & c_2 \\ c_2 & c_1 \end{bmatrix}, \text{ and } M_d = \begin{bmatrix} d_1 & d_2 \\ d_2 & d_1 \end{bmatrix},$$

where

$$a_1 = \frac{1-\Gamma_{i-1}}{L_i} \ , \ a_2 = \frac{(1-\Gamma_{i-1})\Gamma_{si}}{L_i} \ ,$$

$$b_1 = (1-\Gamma_i)\left(\frac{\Gamma_{i-1}}{L_i^2} + \Gamma_{di}(\pi)\right) \ , \ b_2 = (1-\Gamma_i)\left(\frac{2\Gamma_{i-1}\Gamma_{si}}{L_i^2} + \Gamma_{di}(0)\right) \ ,$$

$$c_1 = (1-\Gamma_{i-1})\left(\frac{\Gamma_i}{L_i^2} + \Gamma_{di}(\pi)\right) \ , \ c_2 = (1-\Gamma_{i-1})\left(\frac{2\Gamma_i\Gamma_{si}}{L_i^2} + \Gamma_{di}(0)\right) \ ,$$

$$d_1 = \frac{1-\Gamma_i}{L_i} \ , \ d_2 = \frac{(1-\Gamma_i)\Gamma_{si}}{L_i} \ ,$$

Γ_{i-1} and Γ_i are the power reflectivity for the boundary i-1 and i respectively, L_i is the one way power loss factor for layer i, Γ_{si} is the one way power reflectivity in the specular direction caused by bubbles in layer i, $\Gamma_{di}(\phi)$ is the one way power reflectivity caused by direct scattering from bubbles in ϕ direction in layer i.

For convenience in the multi-layer treatment, the following matrix for each layer is calculated

$$\begin{bmatrix} \tilde{D}_{i-1} \\ \tilde{U}_{i-1} \end{bmatrix} = \begin{bmatrix} M_a^{-1} & -M_a^{-1}M_b \\ M_c M_a^{-1} & M_d - M_c M_a^{-1} M_b \end{bmatrix} \begin{bmatrix} \tilde{D}_i \\ \tilde{U}_i \end{bmatrix} . \qquad (4.36)$$

In order to obtain the relation between the incident and scattering intensities at the air-ice boundary, this matrix for each layer is derived and multiplied, and finally, the matrix for entire ice layer K derived according to

$$\begin{bmatrix} \tilde{D}_0 \\ \tilde{U}_0 \end{bmatrix} = K \begin{bmatrix} \tilde{D}_n \\ \tilde{U}_n \end{bmatrix} = \begin{bmatrix} k_{lm} \end{bmatrix} \begin{bmatrix} \tilde{D}_n \\ \tilde{U}_n \end{bmatrix} , \qquad (4.37)$$

where k_{lm} is the element of the matrix in row l and column m. Because there is no

upward intensity from the bottom layer (water or frozen soil) and no left downward intensity in the upper layer (air), the following conditions must be satisfied:

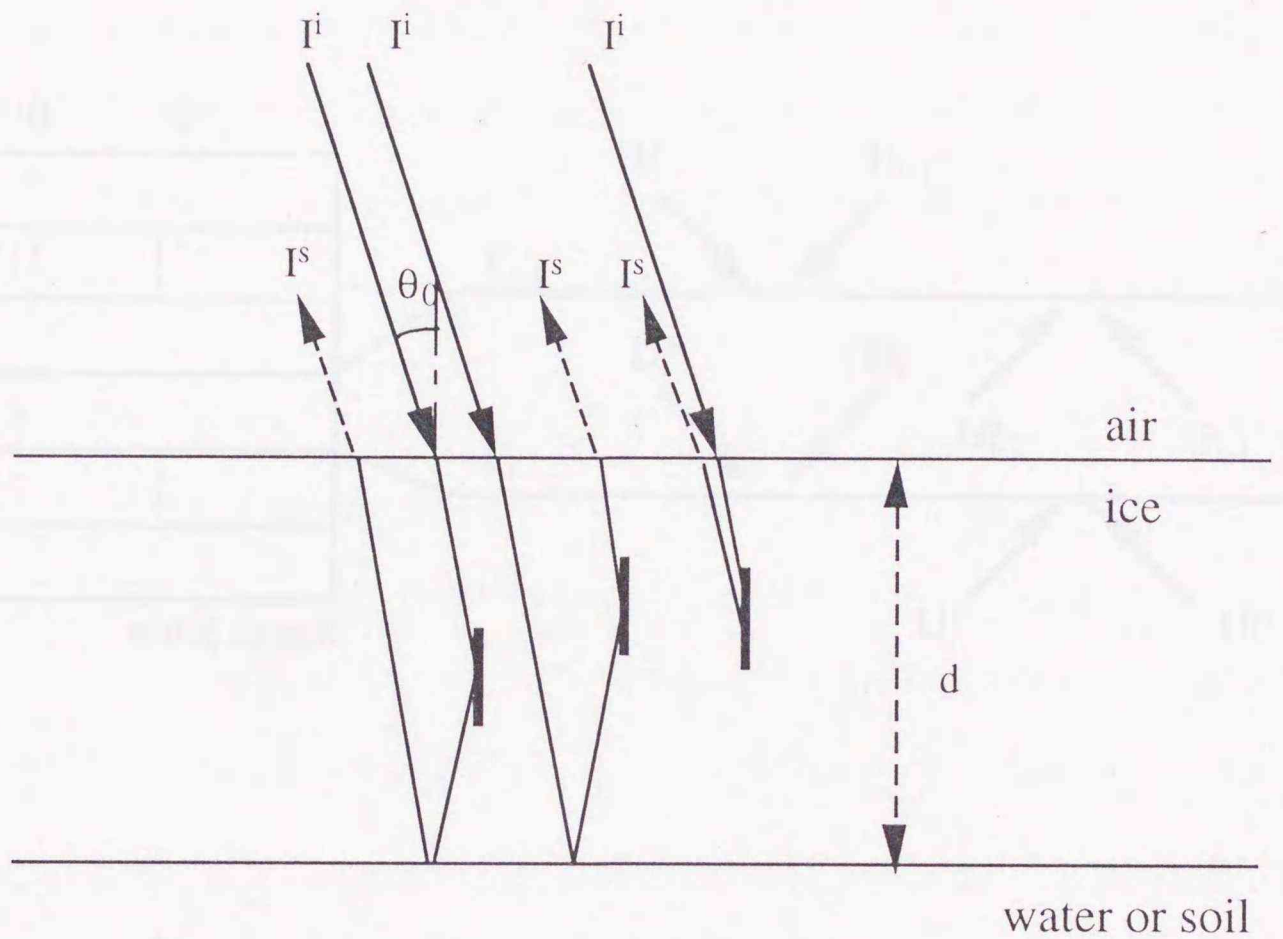
$$D_0^+ = I^i, D_0^- = U_n^+ = U_n^- = 0. \tag{4.38}$$

Using the above conditions, the ratio of backscattering intensity to the incident intensity is given by

$$\frac{I^s}{I^i} = (1-\Gamma_0)\cdot\frac{U_0^-}{D_0^+} = (1-\Gamma_0)\cdot\frac{k_{41}-\dfrac{k_{42}k_{21}}{k_{22}}}{k_{11}-\dfrac{k_{12}k_{21}}{k_{22}}}. \tag{4.39}$$

Substituting the equation (4.39) to (4.34), finally we obtain the backscattering coefficient as follows [*Wakabayashi et al.* , 1993]:

$$\sigma^0 = 4\pi\cos\theta_0\cdot(1-\Gamma_0)\cdot\frac{k_{41}-\dfrac{k_{42}k_{21}}{k_{22}}}{k_{11}-\dfrac{k_{12}k_{21}}{k_{22}}}. \tag{4.40}$$



θ_0 :incidence and scattered angle
d : ice thickness
I^i :incident wave
I^s :scattered wave

Fig. 4-5 Schematic figure of lake ice reflection including direct backscattering and forward scattering from bubbles. The upper and bottom boundaries of the ice are considered to be specular surfaces.

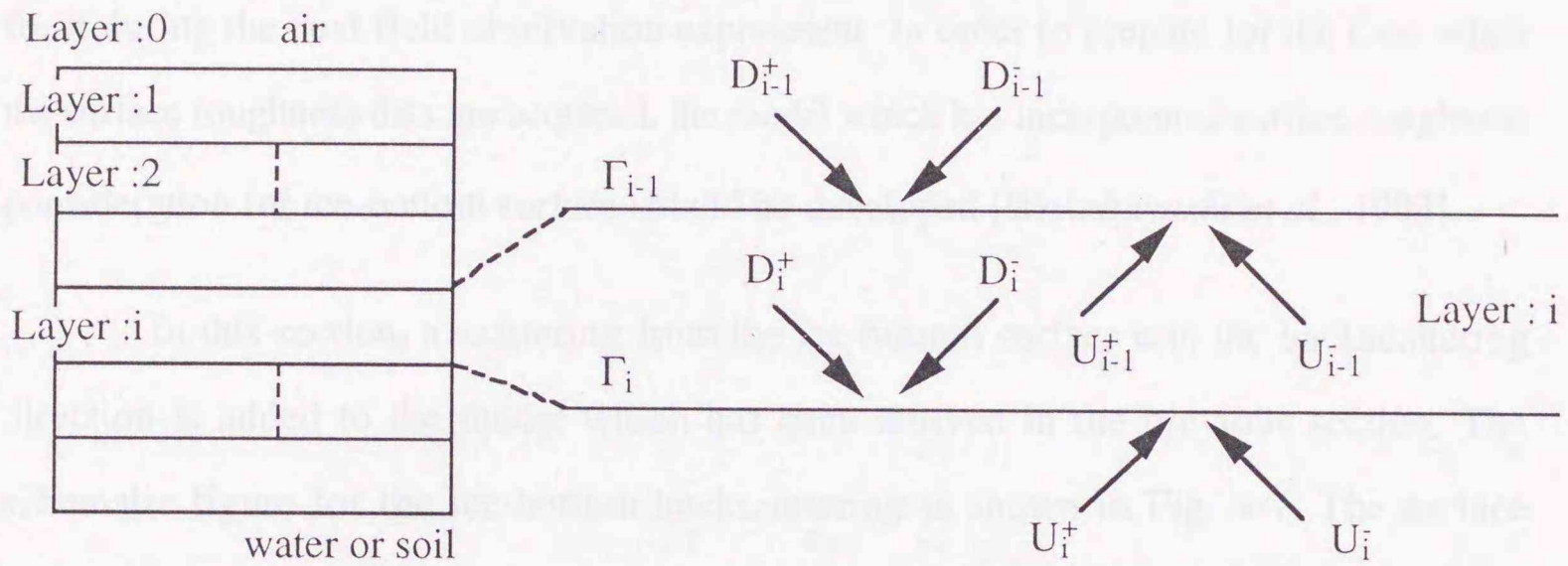


Fig. 4-6 Multi-layer ice structure. The ice is composed of n sub-layers with each sub-layer having a different bubble density, bubble shape and bubble size.

4-7. SURFACE SCATTERING FROM ICE-BOTTOM BOUNDARY

In the pervious section, the ice-bottom boundary was considered to be a specular surface and the ice model was made based on this assumption. Since there is no observation data available for the ice-bottom surface roughness now, we need to make a measurement for it during the next field observation experiment. In order to prepare for the time when the surface roughness data are acquired, the model which has incorporated surface roughness consideration for ice-bottom surface should be developed [*Wakabayashi et al.*, 1993].

In this section, a scattering from the ice-bottom surface into the backscattering direction is added to the model which has been derived in the previous section. The schematic figure for the ice-bottom backscattering is shown in Fig. 4-7. The surface roughness is characterized by two parameters, one is the standard deviation of surface height σ_ℓ, and another is the surface correlation length ℓ [*Ulaby et al.*, 1982]. We assume that both σ_ℓ and ℓ are much smaller than the wavelength in the case of the ice bottom surface on the lakes. For a slightly rough surface, a standard approach for scatter modeling is the small-perturbation model, which requires the surface standard deviation to be less than about 5 percent of the wavelength and the average slope of the surface to be of the same order as the wave number times the surface standard deviation. The above conditions are mathematically given by

$$k\sigma_\ell < 0.3, \tag{4.41}$$

$$\frac{\sqrt{2}\sigma_\ell}{\ell} < 0.3. \tag{4.42}$$

The backscattering coefficient of the small perturbation model $\sigma^0_{spm_pq}$ shown in Fig. 4-7, which corresponds to a q-polarized incident wave that gives rise to a p-polarized wave is given as follows [*Ulaby et al.*, 1982]:

$$\sigma^0_{spm_pq} = 8k^4\sigma_\ell^2 \cos^4\theta \left|\alpha_{pq}\right|^2 W(2k\sin\theta, 0), \tag{4.43}$$

where $\alpha_{hh} = R_H$ for HH polarization, which is the Fresnel reflection coefficient for H polarization,

$$\alpha_{vv} = (\frac{\varepsilon_2}{\varepsilon_1} - 1)\frac{\sin^2\theta - \frac{\varepsilon_2}{\varepsilon_1}(1+\sin^2\theta)}{\left[\frac{\varepsilon_2}{\varepsilon_1}\cos\theta + (\frac{\varepsilon_2}{\varepsilon_1} - \sin^2\theta)^{\frac{1}{2}}\right]^2} \text{ for VV polarization,}$$

$$\alpha_{hv} = \alpha_{vh} = 0 \text{ for HV and VH polarization,}$$

and $W(2k\sin\theta, 0)$ is the Fourier transformation of the surface correlation coefficient, which is called the normalized roughness spectrum. When a Gaussian correlation coefficient of the form $\exp(-\xi^2/\ell^2)$ is assumed, which corresponds to an isotropic roughness spectrum, its Fourier transformation of the surface correlation coefficient is given as follows [*Ulaby et al.*, 1982],

$$W(2k\sin\theta) = \frac{1}{2}\ell^2 \exp\left[-(k\ell\sin\theta)^2\right]. \tag{4.44}$$

In order to incorporate ice-bottom surface scattering into the lake ice model, the reflectivity of the ice-bottom surface is derived as

$$\Gamma_{spm_pq}(\theta) = \frac{1}{4\pi\cos\theta}\sigma^0_{spm_pq}, \tag{4.45}$$

and added to the matrix in Equation (4.35) for sub-layer n (the bottom layer in the ice). Strictly speaking, the effect of surface roughness on the specular reflected component should be considered by multiplying the Fresnel reflection coefficient by a factor that depends on the surface roughness parameters, but the effect of this component is considered to be very small, so this effect will not be included in the proposed model.

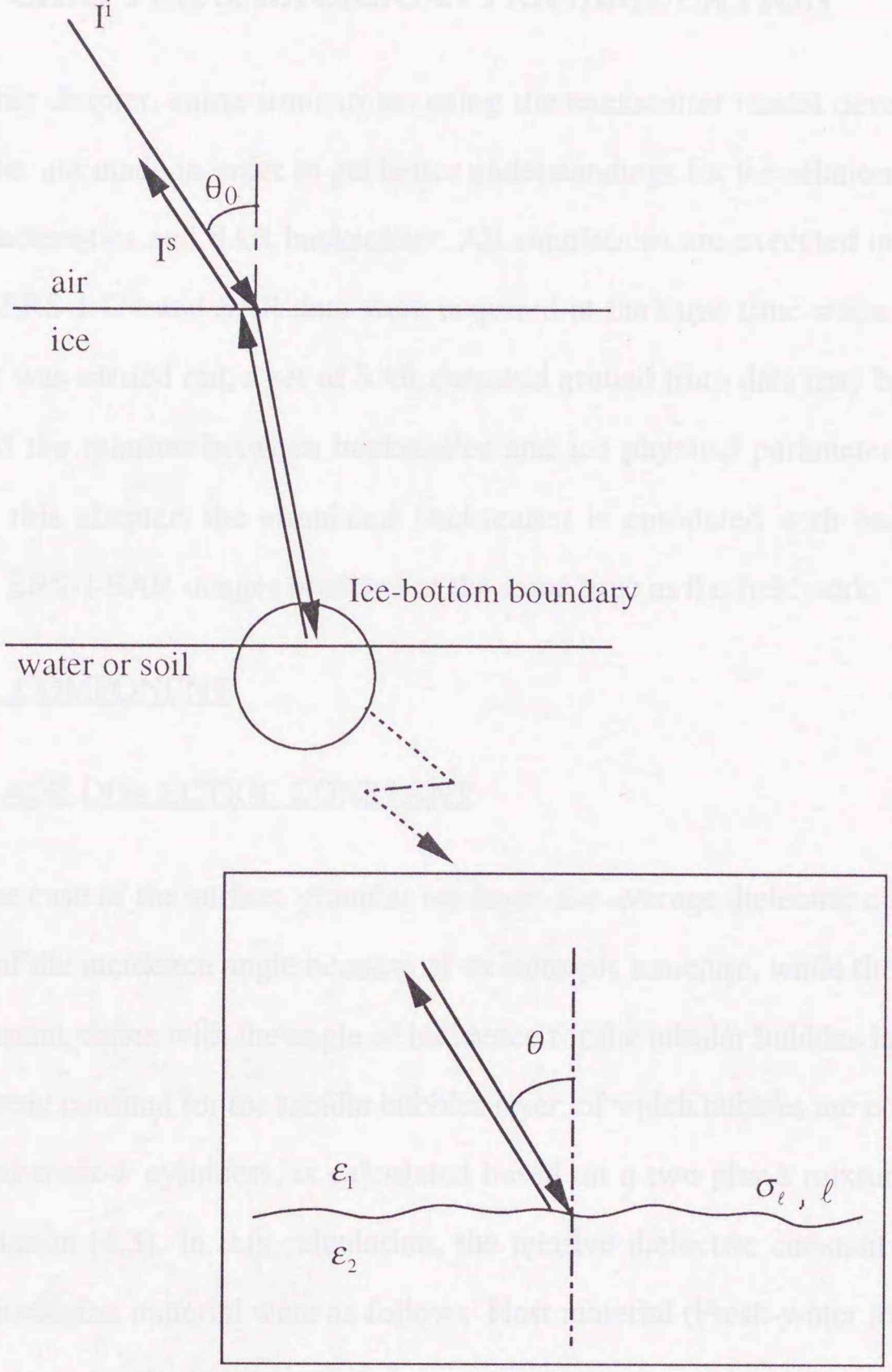


Fig. 4-7 Backscattering from rough ice-bottom boundary.

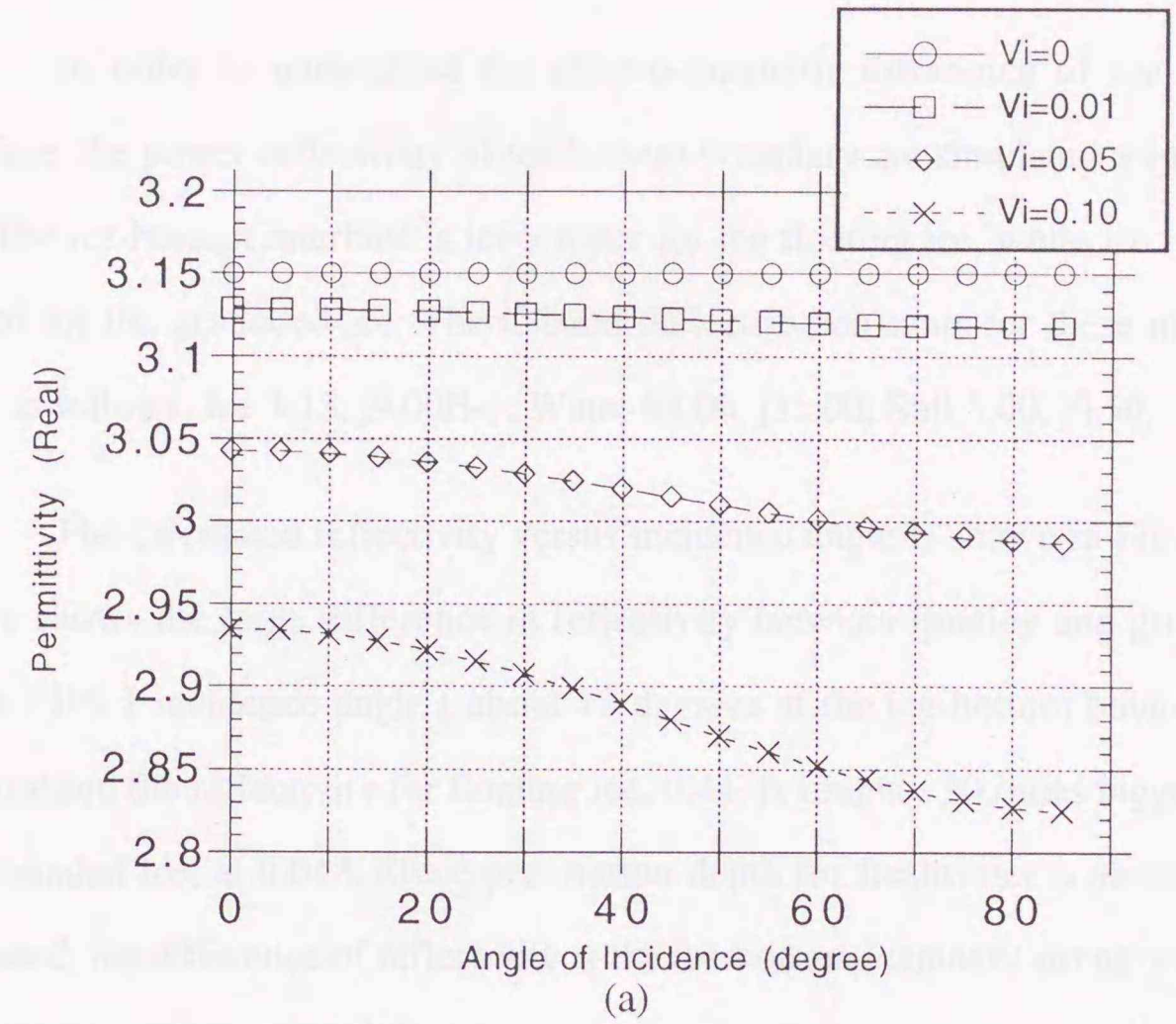
CHAPTER 5. BACKSCATTER SIMULATION

In this chapter, some simulations using the backscatter model developed in the last chapter are made in order to get better understandings for the relation between lake ice characteristics and SAR backscatter. All simulations are executed in C-band. Because the ERS-1 C-band SAR data were acquired at the same time when the field measurement was carried out, a set of SAR data and ground truth data may be helpful to understand the relation between backscatter and ice physical parameters. In the final part of this chapter, the simulated backscatter is compared with backscatter derived from ERS-1 SAR images obtained at the same time as the fieldwork.

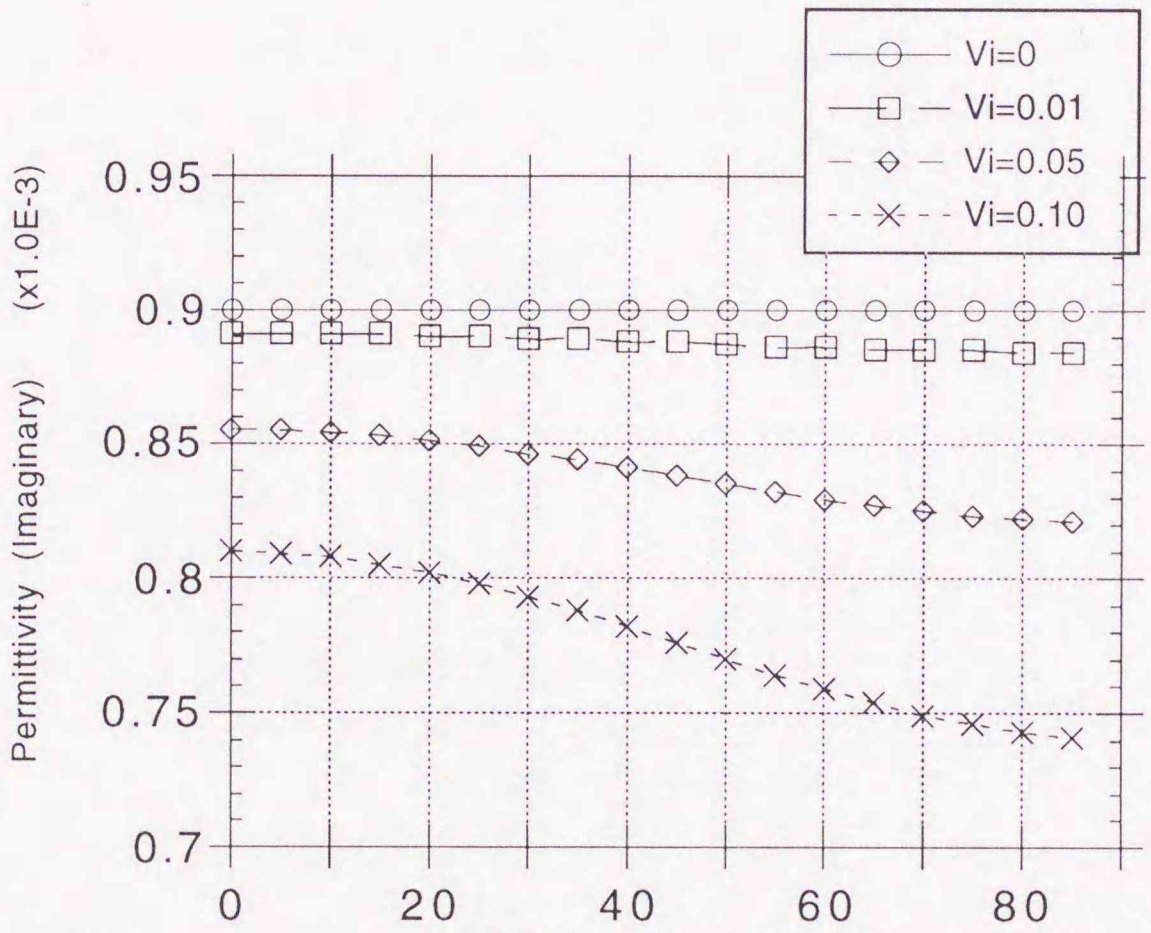
5-1. MODEL COMPONENT

5-1.1. AVERAGE DIELECTRIC CONSTANT

In the case of the surface granular ice layer, the average dielectric constant is independent of the incidence angle because of its isotropic structure, while the average dielectric constant varies with the angle of incidence for the tubular bubbles layer. The average dielectric constant for the tubular bubbles layer, of which bubbles are considered to be long and narrow cylinders, is calculated based on a two-phase mixture model given by Equation (4.3). In this calculation, the relative dielectric constant for host material and inclusion material were as follows: Host material (Fresh-water ice) 3.15 - j 9.00E-4 ; Inclusion material (Air) 1.00 - j 0 . The volume inclusion factor v_i was set in four cases, such as 0, 0.01, 0.05, and 0.1. Calculated real and imaginary dielectric constants versus incidence angle are plotted in Fig. 5-1. Since the change of the dielectric constants with the volume inclusion factor is found to be very small, the reflection from the boundary between ice sub-layers can be considered to be very small according to Fresnel's law.



(a)



(b)

Fig. 5-1 Average dielectric constant versus incidence angle calculated by two-phase mixture model: (a) Real part of dielectric constant; (b) Imaginary part of dielectric constant.

5-1.2. REFLECTIVITY AT ICE-BOTTOM BOUNDARY

In order to understand the electro-magnetic difference of the ice-bottom interface, the power reflectivity of ice-bottom boundary are simulated using Fresnel's law. The ice-bottom interface is ice - water for the floating ice, while ice - frozen soil is used for the grounded ice. The C-band dielectric constants for these materials are taken as follows: Ice 3.15, j9.00E-4; Water 65.00, j35.00; Soil 5.00, j0.50.

The calculated reflectivity versus incidence angle is shown in Fig. 5-2 which clearly shows the large difference in reflectivity between floating and grounded ice. At the ERS-1 incidence angle (about 13 degrees at the ice-bottom boundary) with polarization, the reflectivity for floating ice, 0.44, is roughly 30 times bigger than that for grounded ice, at 0.013. Since penetration depth for freshwater is about 18 meters in C-band, the difference of reflectivity at the ice-bottom boundary strongly contributes to the backscatter from the lake ice.

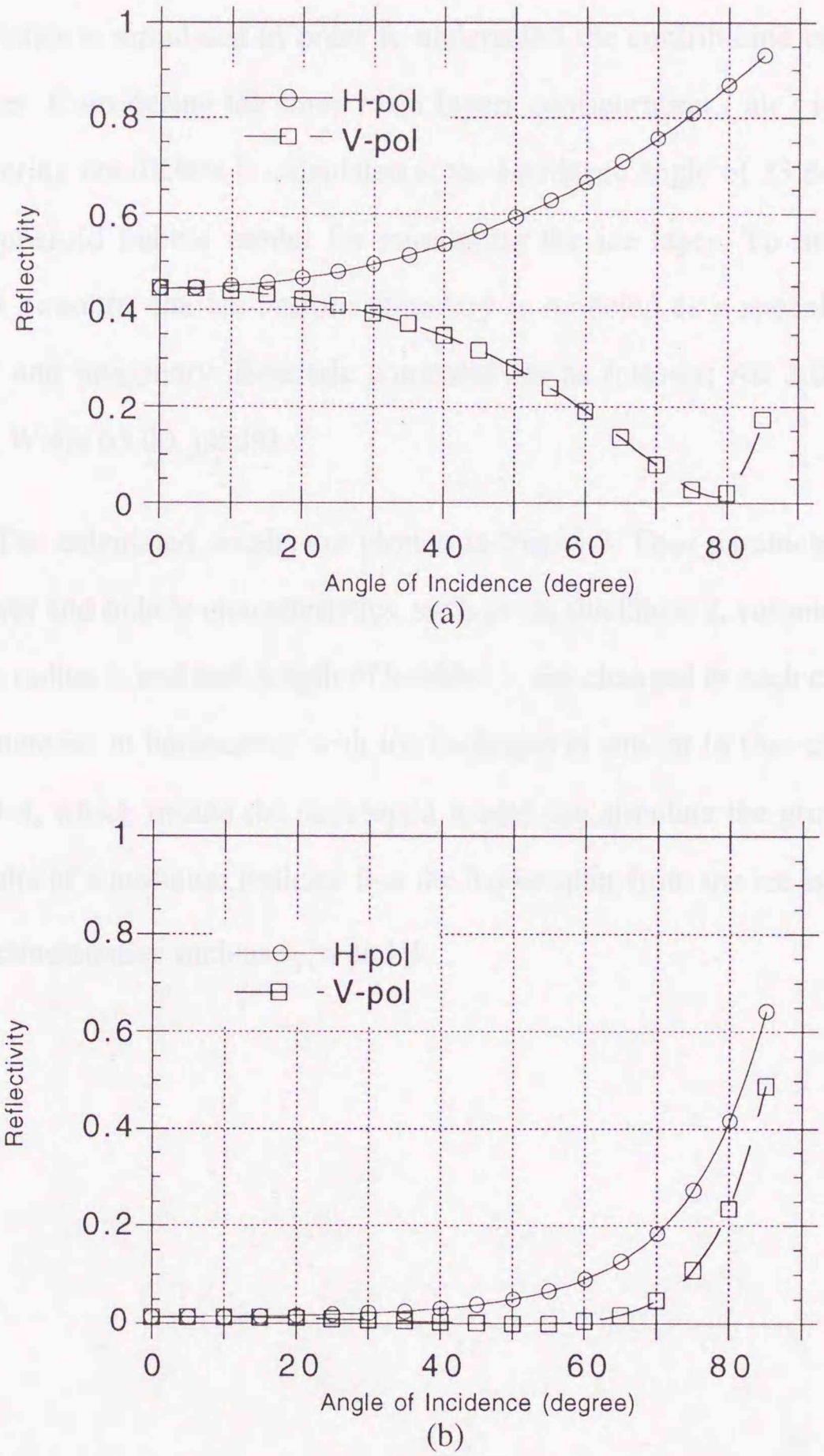
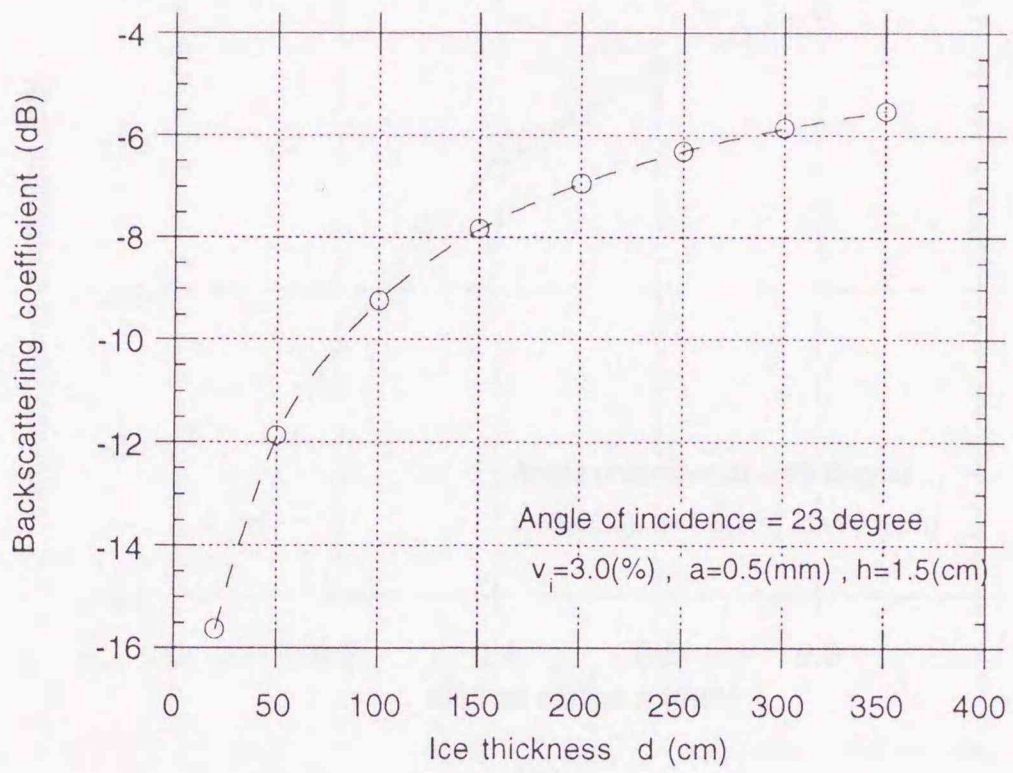


Fig. 5-2 Power reflectivity at ice-bottom boundary versus incidence angle calculated by Fresnel's law: (a) Floating ice; (b) Grounded ice.

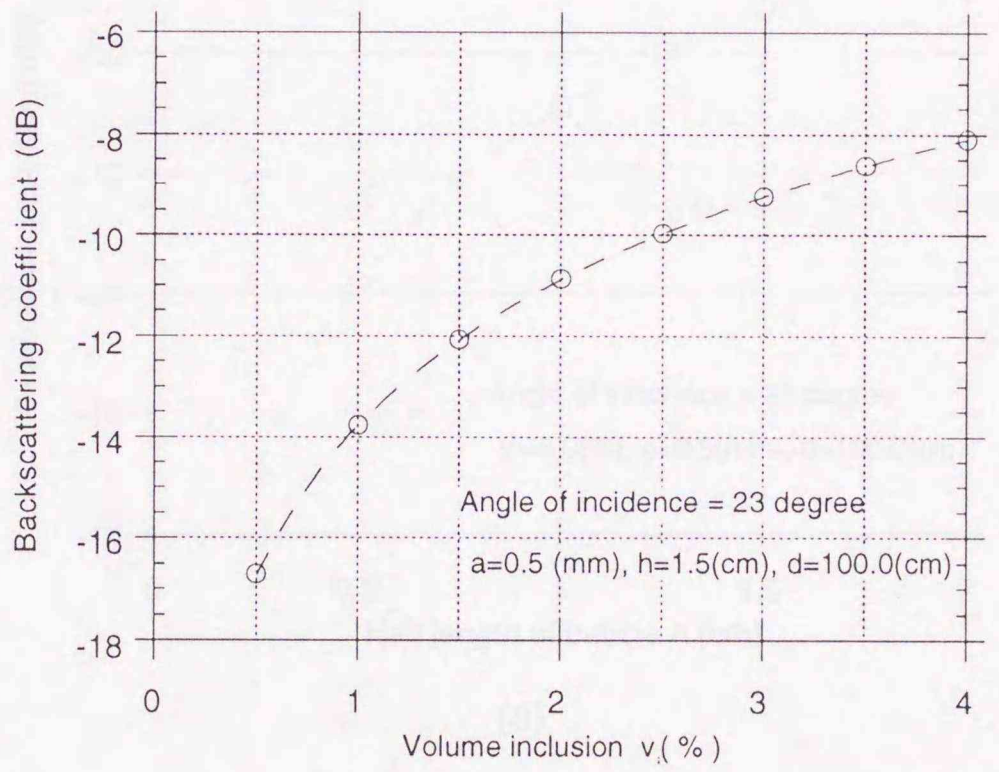
5-1.3. BUBBLY ICE LAYER

The backscatter from simple one layer ice which has various bubble characteristics is simulated in order to understand the contribution of bubbles to the backscatter. Considering the three main layers configuration (air / ice / water), the backscattering coefficient is calculated at the incidence angle of 23 degrees using the prolate spheroid bubble model for simulating the ice layer. To make a model as simple as possible, the ice-bottom boundary is modeled as a specular surface. The used real and imaginary dielectric constants are as follows; Air 1.00, j0; Ice 3.15, j9.00E-4; Water 65.00, j35.00.

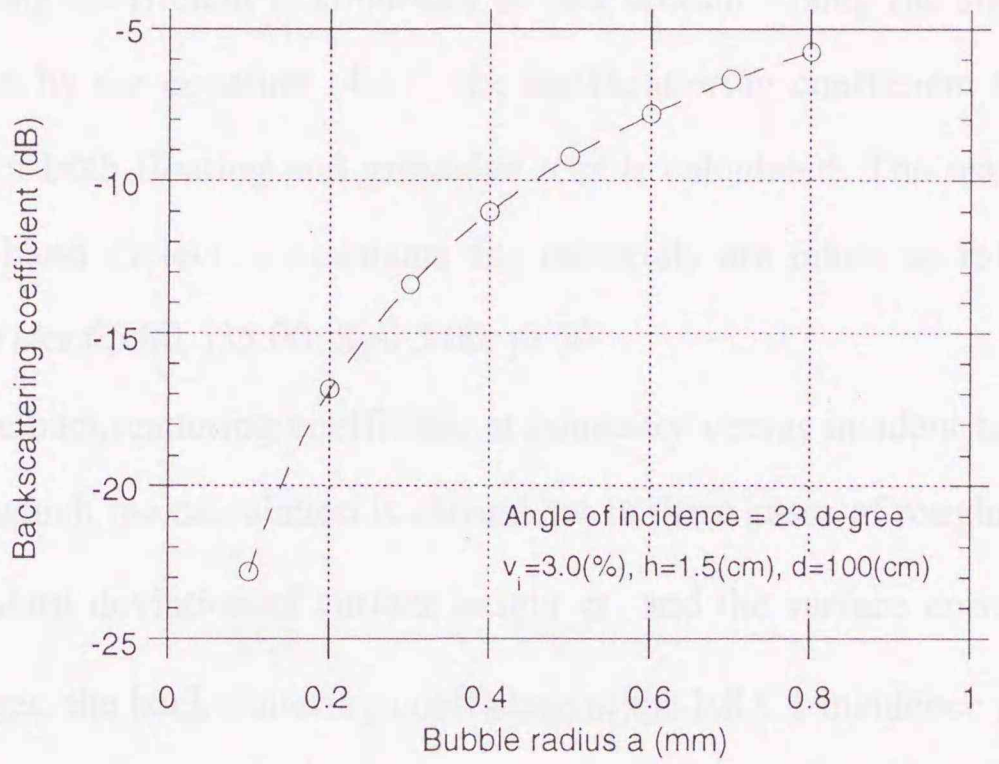
The calculated results are plotted in Fig. 5-3. Four parameters representing the ice layer and bubble characteristics, such as ice thickness d, volume inclusion ratio v_i, bubble radius a, and half length of bubbles h, are changed in each calculation. Note that the increase in backscatter with ice thickness is similar to that observed in Figs. 3-3 and 3-4, which means the developed model can simulate the growth of ice. The other results of simulation indicate that the backscatter from the ice is sensitive to the bubble characteristics such as v_i, a and h.


Backscattering coefficient (dB)
-4
-6
-8
-10
-12
-14
-16
0
50
100
150
200
250
300
350
400
Ice thickness d (cm)
Angle of incidence = 23 degree
v_i=3.0(%) , a=0.5(mm) , h=1.5(cm)

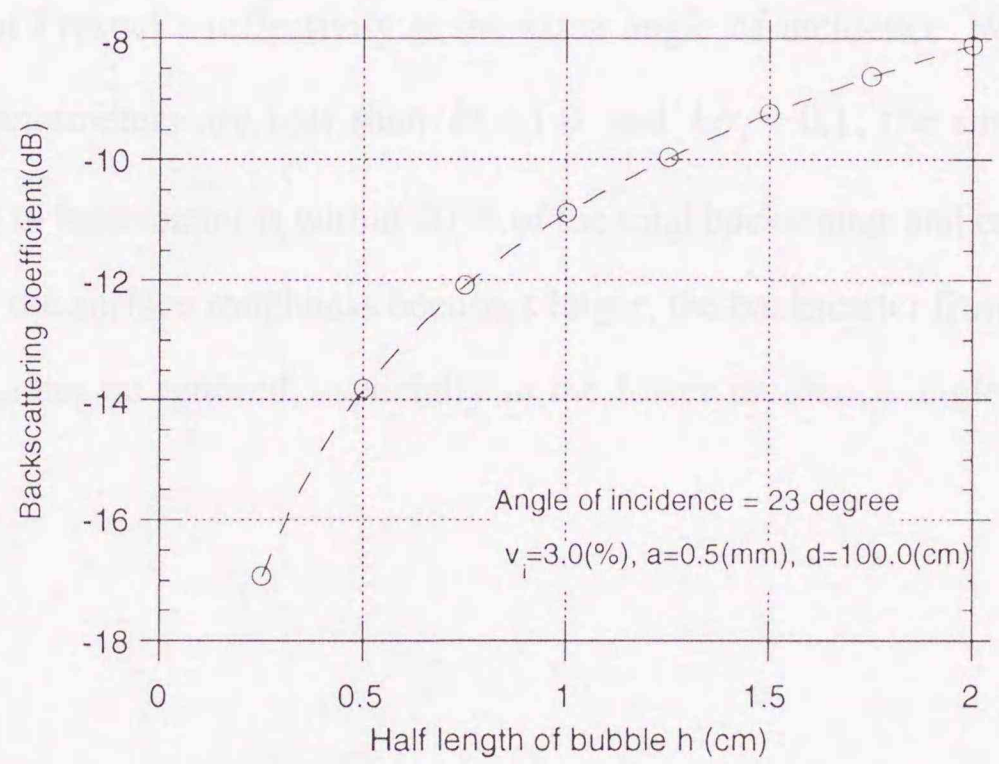
(a)


Backscattering coefficient (dB)
-6
-8
-10
-12
-14
-16
-18
0
1
2
3
4
Volume inclusion v_i (%)
Angle of incidence = 23 degree
a=0.5 (mm), h=1.5(cm), d=100.0(cm)

(b)



(c)



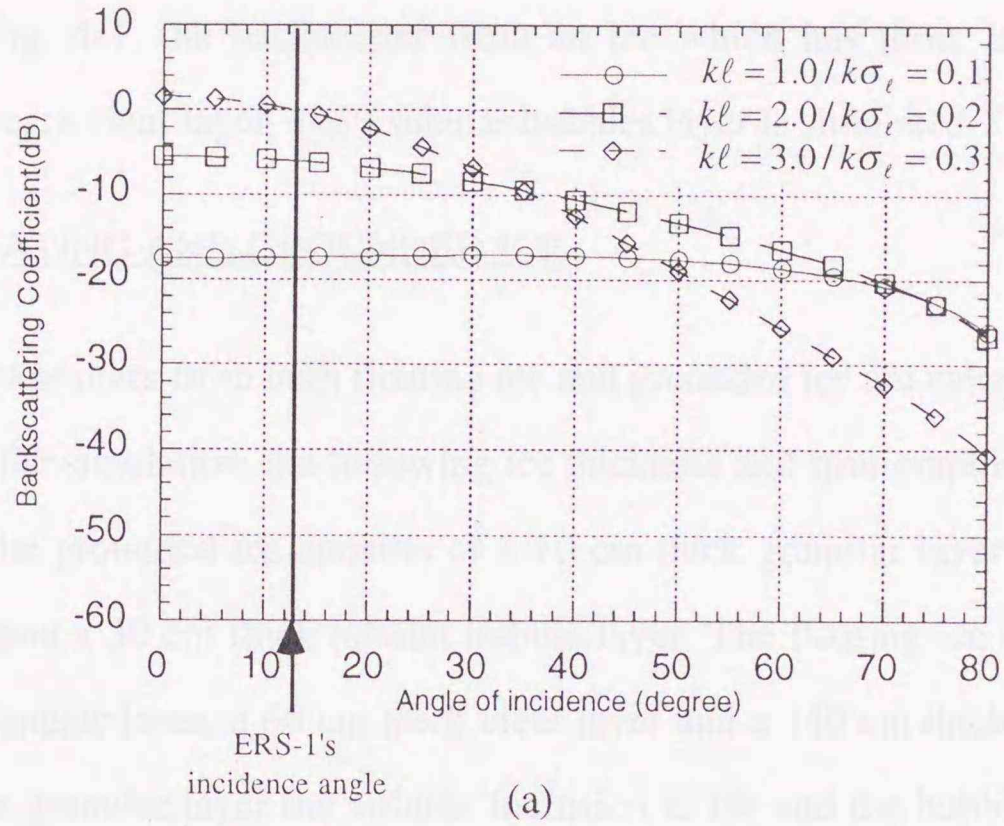
(d)

Fig. 5-3. Backscattering coefficient at C-band (VV-polarization) versus each parameter characterizes tubular bubbles layer: (a) σ^0 vs. ice thickness; (b) σ^0 vs. the volume inclusion factor of the air; (c) σ^0 vs. bubble radius; and (d) σ^0 vs. half bubble length. For each case, the other conditions are as indicated in the figures.

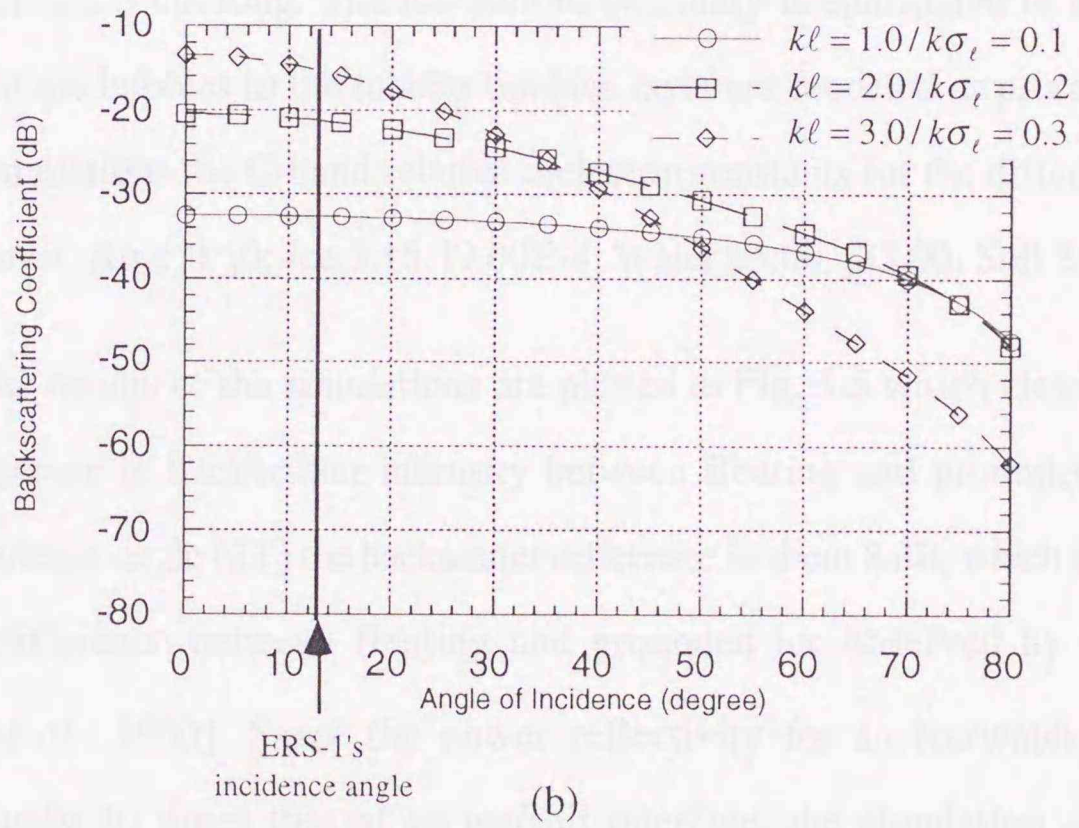
5-1.4. SURFACE ROUGHNESS AT ICE-BOTTOM BOUNDARY

The contribution of surface roughness at the ice-bottom boundary to the backscattering coefficient is simulated in this section. Using the small perturbation model given by the equation (4.43), the backscattering coefficient from ice-bottom boundary for both floating and grounded ices is calculated. The real and imaginary parts of C-band dielectric constants for materials are taken as follows: Ice 3.15, j9.00E-4; Water 65.00, j35.00; Soil 5.00, j0.50.

The backscattering coefficient at boundary versus incident angle is shown in Fig. 5-4 in which the calculation is carried out in three cases of roughness parameters. As the standard deviation of surface height σ_ℓ and the surface correlation length ℓ become larger, the backscattering coefficient at the ERS-1 incidence angle (about 13 degree at ice-bottom boundary) is significantly increased. As shown in Fig. 5-4, the difference of backscattering coefficient between the grounded ice and the floating ice is about 15 dB at 13 degree of angle of incidence, which is comparable with the difference of Fresnel's reflectivity at the same angle of incidence. When the surface roughness parameters are less than $k\ell = 1.0$ and $k\sigma_\ell = 0.1$, the surface roughness contribution to backscatter is within 10 % of the total backscatter and can be negligible. However, if the surface roughness becomes larger, the backscatter from the ice-bottom boundary cannot be ignored, especially in the lower incidence angle, and should be included.



(a)



(b)

Fig. 5-4 Backscattering coefficient at the ice-bottom boundary versus incidence angle calculated by using the small perturbation model: (a) Floating ice in polarization VV; (b) Grounded ice in polarization VV.

5-2. MULTI-LAYER SIMULATION

Considering the typical structure of lake ice as observed in Northern Alaska shown in Fig. 4-1, the backscatter from an ice which has three layers such as a granular layer, a clear layer, and a tubular bubbles layer is simulated.

5-2.1. FLOATING AND GROUNDED ICE

Backscatters from both floating ice and grounded ice are calculated using this model. For the simulation, the following ice thickness and stratigraphic characteristics are used. The grounded ice consists of a 10 cm thick granular layer, a 15 cm thick clear layer and a 30 cm thick tubular bubble layer. The floating ice consists of a 10 cm thick granular layer, a 60 cm thick clear layer and a 140 cm thick tubular bubble layer. In the granular layer the volume inclusion is 1% and the bubble diameter is 2 mm, while in the tubular layer the volume inclusion is 3% and the bubbles are 1 mm in diameter and 3 cm long. The ice-bottom boundary is considered to be a specular surface and the bubbles in the tubular bubbles layer are modeled as prorate spheroids. In these simulations the C-band relative dielectric constants for the different materials are as follows: Air 1.0, j0; Ice 3.15, j9.00E-4; Water 65.00, j35.00; Soil 5.00, j0.50.

The results of the simulations are plotted in Fig. 5-5 which clearly shows the large difference in backscatter intensity between floating and grounded ice. At the ERS-1 incidence angle (23°) the backscatter difference is about 8 dB, which is comparable with the difference between floating and grounded ice observed by ERS-1 SAR [Jeffries *et al.*, 1993]. Since the power reflectivity for an ice/water interface is approximately 30 times that of an ice/soil interface, the simulation confirms that backscatter is significantly changed by the presence or absence of a specular ice/water interface.

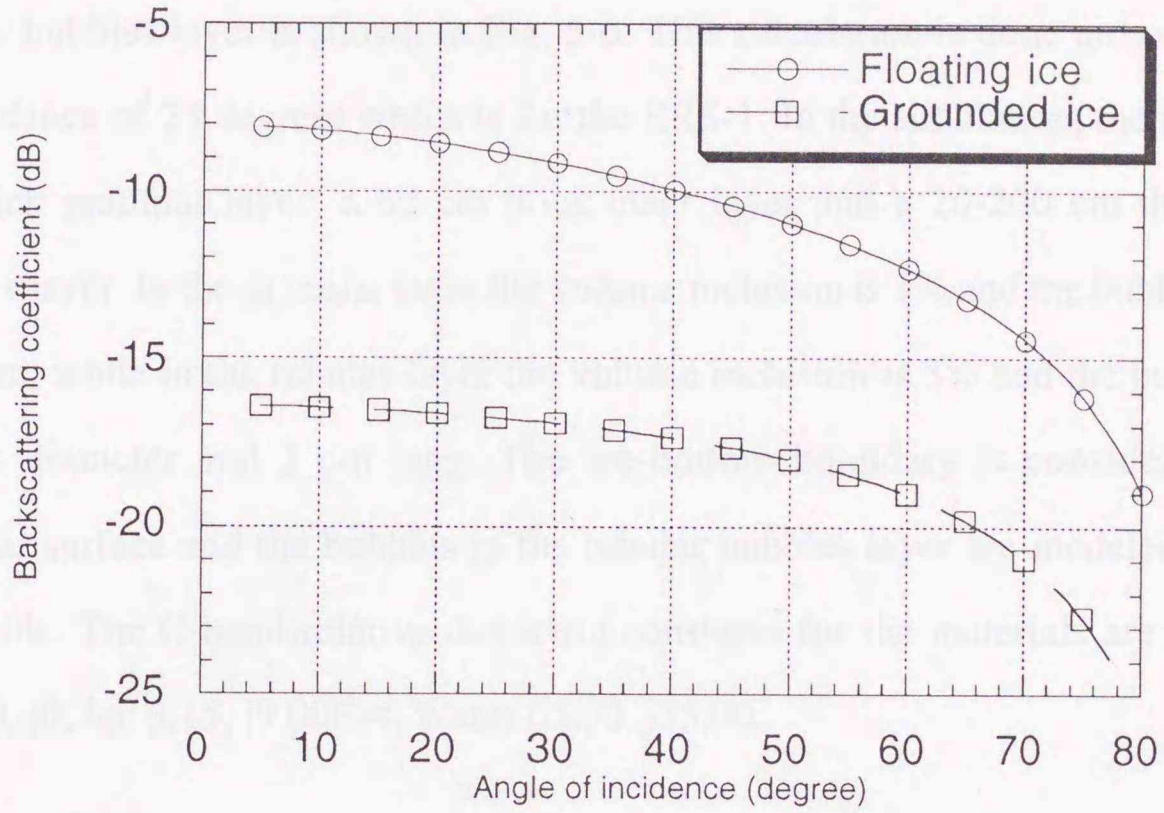


Fig. 5-5 Backscattering coefficient at C-band (VV) versus angle of incidence for floating ice and grounded ice. Each ice has three layers such as a granular layer, a clear layer and a tubular bubbles layer.

5-2.2. GROWING ICE

A backscatter change for the floating ice as a function of ice thickness in the tubular bubbles layer is shown in Fig. 5-6. This calculation is done only at the angle of incidence of 23 degrees which is for the ERS-1. In the simulation, the ice has a 10 cm thick granular layer, a 60 cm thick clear layer and a 20-200 cm thick tubular bubbles layer. In the granular layer the volume inclusion is 1% and the bubble diameter is 2 mm, while in the tubular layer the volume inclusion is 3% and the bubbles are 1 mm in diameter and 3 cm long. The ice-bottom boundary is considered to be a specular surface and the bubbles in the tubular bubbles layer are modeled as prorate spheroids. The C-band relative dielectric constants for the materials are as follows: Air 1.0, j0; Ice 3.15, j9.00E-4; Water 65.00, j35.00.

The increase in backscatter with increasing ice thickness, which is ranging from -16 dB to -7.5 dB, is comparable with that observed by SAR during the period from October 1991 to April 1992 shown in Fig. 3-3.

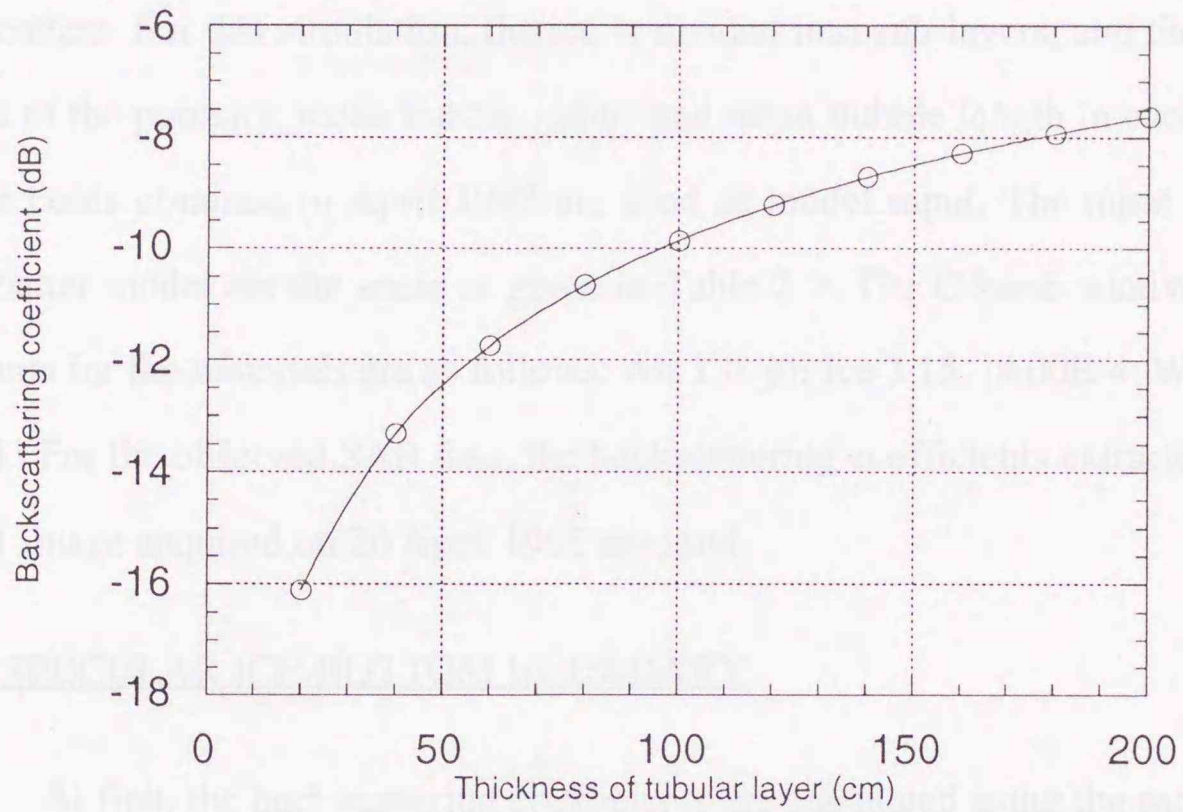


Fig. 5-6 Backscattering coefficient at C-band (VV) versus the thickness of an ice layer with tubular bubbles.

5-3. COMPARISON BETWEEN ERS-1 SAR AND MODELED BACKSCATTER

The final part of this chapter compares the simulated and the observed backscatters. For this simulation, the ice is divided into sub-layers, and the measured values of the porosity, mean bubble radius and mean bubble length in each sub-layer for ice cores obtained in April 1992 are used as model input. The input data to the backscatter model are the same as given in Table 2-2. The C-band relative dielectric constants for the materials are as follows: Air 1.0, j0; Ice 3.15, j9.00E-4; Water 65.00, j35.00. For the observed SAR data, the backscattering coefficients extracted from the ERS-1 image acquired on 20 April 1992 are used.

5-3.1. SPECULAR ICE-BOTTOM BOUNDARY

At first, the backscattering coefficients are calculated using the model having a specular ice-bottom boundary and compared with the backscatter derived from the actual SAR data. The result of the comparison between SAR-derived backscatter and the model output is shown in Fig. 5-7. At most sites, the model underestimates the backscatter, particularly at sites 7, 8 and 11. The correlation coefficient for a linear regression of the two data sets is 0.73 and the model has -5.5 dB offset relative to the SAR data.

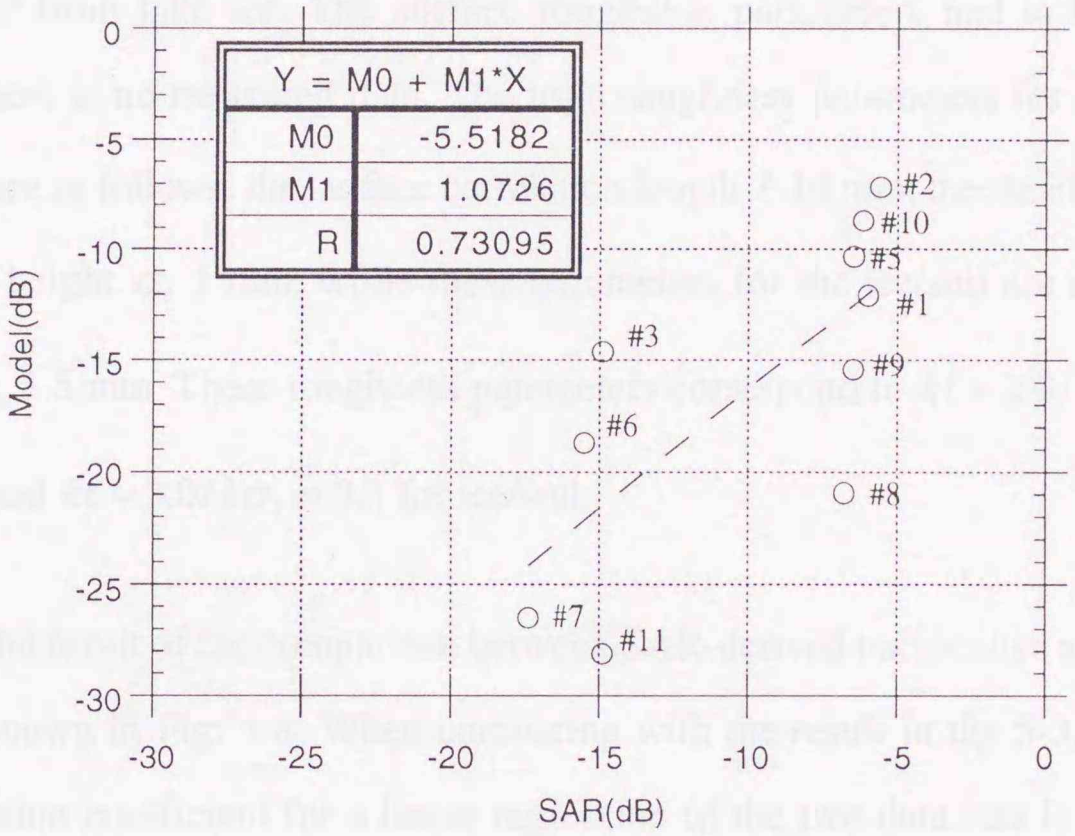


Fig. 5-7 The backscattering coefficient observed by ERS-1 SAR versus the backscattering coefficient estimated by lake ice model with a specular ice-bottom boundary. Numbers displayed on the figure indicate the site number. The linear dotted line is the result of linear curve fitting.

5-3.2. ROUGH ICE-BOTTOM BOUNDARY

The model which has the rough ice-bottom boundary is used to simulate backscatter from lake ice. The surface roughness parameters had to be estimated because there is no measured data. The used roughness parameters for the ice/water boundary are as follows: the surface correlation length ℓ 10 mm; the standard deviation of surface-height σ_ℓ 1 mm, while these parameters for the ice/soil are as follows: ℓ 15 mm.; σ_ℓ 1.5 mm. These roughness parameters correspond to $k\ell = 2.0 / k\sigma_\ell = 0.2$ for ice/water and $k\ell = 3.0 / k\sigma_\ell = 0.3$ for ice/soil.

The result of the comparison between SAR-derived backscatter and the model output is shown in Fig. 5-8. When comparing with the result in the 5-3.1 (Fig. 5-7), the correlation coefficient for a linear regression of the two data sets is improved to 0.97. Although the roughness parameters used in this calculation are not a measured value, by adding consideration for the surface roughness at the ice-bottom boundary, the model backscatter remarkably approaches the observed SAR backscatter. In other words, for the backscattering model of lake ice in northern Alaska, the surface roughness of ice-bottom boundary should be included.

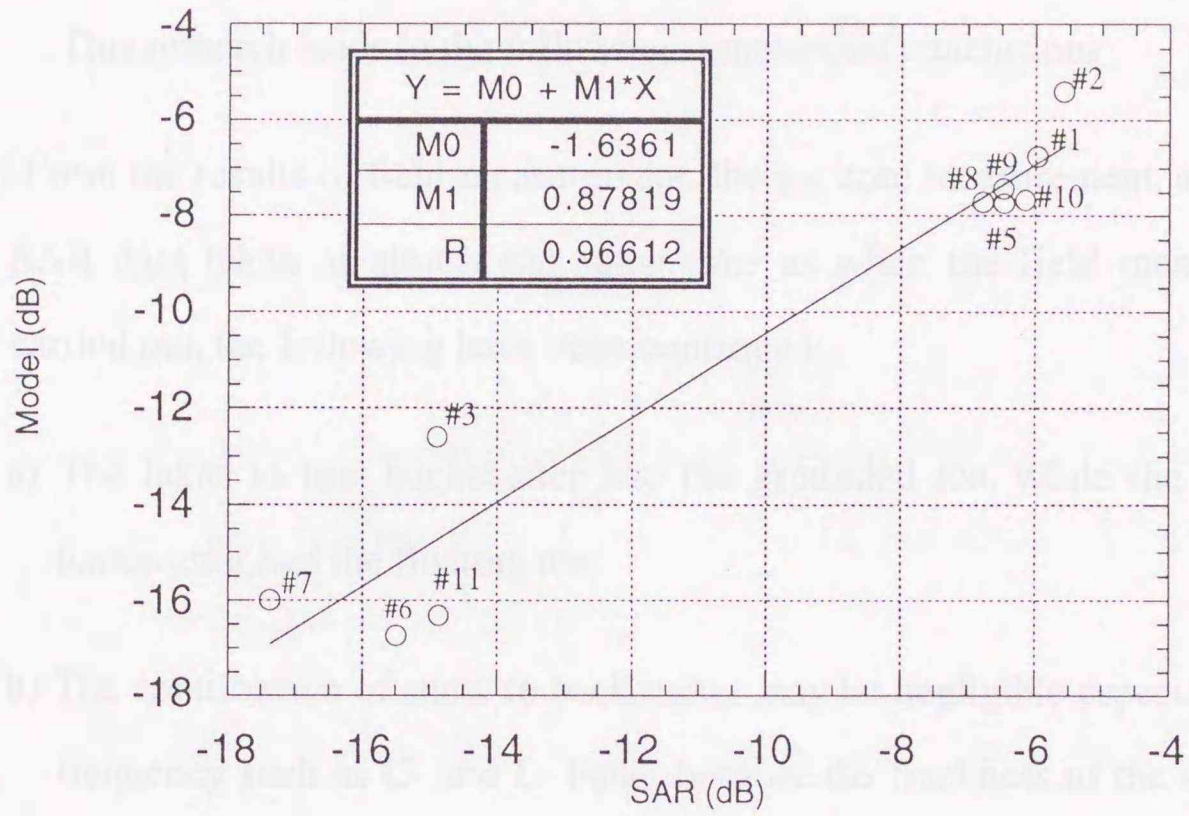


Fig. 5-8 The backscattering coefficient observed by ERS-1 SAR versus the backscattering coefficient estimated by the lake ice model with a rough ice-bottom boundary. Numbers displayed on the figure indicate the site number. The linear line is the result of linear curve fitting.

CHAPTER 6. CONCLUSIONS

This research leads to the following summarized conclusions.

(1) From the results of field measurement, the ice core measurement, and the C-band SAR data taken at almost the same time as when the field measurement was carried out, the following have been confirmed:

a) The lakes in low backscatter had the grounded ice, while the lakes in high backscatter had the floating ice;

b) The contribution of snow to backscatter may be negligible especially in the low frequency such as C- and L- band, because the thickness of the snow cover on the ice was equal to or less than 80 mm at the all sites and the snow itself was very dry;

c) No difference was found in the roughness for both snow surface and ice surface at all of the 11 sites;

d) The typical lake ice observed in the measured area was composed of three kinds of layers, which were the granular layer with spherical bubbles, the clear and bubble-free layer, and the tubular bubbles layer. The clear layer and the tubular bubbles layer were found at all of the sites, but the granular layer was found at 6 out of 11 sites. The bubble density and bubble size varied with their depth in the tubular bubbles layer.

(2) The time-series of radiometrically calibrated ERS-1's C-band data of the Barrow, Alaska region acquired and processed at ASF have been analyzed. The results of analyses are as follows;

a) As the ice grows, the backscatter increased from -22 dB (Minimum value in

September) to -6 dB (Maximum value in January) as long as the ice was floating on the water. The increase of backscatter was found about 10 dB at most sites .

b) When the ice was finally grounded, the backscattering coefficient decreased below -15 dB. The backscatter difference between floating ice and grounded ice was 8 dB to 10 dB.

c) Assuming that the ice growing speed is the same in all the lakes, the time series of C-band backscatter from lakes may be helpful for lake batymetry. The change of the backscatter profile proved that the boundary between high backscatter and low backscatter (the boundary point of floating and grounded ice) moved to the center of the lake during the winter time.

(3) The JERS-1's L-band SAR data for the Barrow area have been analyzed. Although there was no field measurement data available, the followings were confirmed by using only radiometrically calibrated SAR data:

a) The backscatter difference between floating and grounded ice was about 2 dB which was much smaller than that in C-band.

b) The increase of backscattering coefficient for growing ice was a maximum of 4 dB approximately.

c) The higher JERS-1's $NE\sigma^0$ is assumed to cause the smaller backscatter difference between the floating ice and the grounded ice in addition to the smaller backscatter increase of growing ice.

(4) Based on the ice characteristics obtained from ice core measurements, a numerical model for backscatter from ice growing on shallow tundra lakes has been developed. First the ice is divided into sub-layers which have different bubble characteristics, and then the first order scattering is calculated in each sub-layer. After scatterers in sub-layers are combined, the relation of incidence and scattering intensities at the

top surface of the ice is calculated and converted into a backscattering coefficient.

(5) Using the developed model, each part which affects the backscattering coefficient was calculated and evaluated. This backscatter model could explain the backscatter difference between grounded ice and floating ice, and also the observed increase in backscatter as the ice grows thicker during the winter.

(6) Backscattering coefficients derived from ERS-1's SAR data and the model output were compared. The model which has a specular ice-bottom boundary underestimated backscatter, but after considering the rough ice-bottom boundary a good agreement between the model output and SAR data was obtained. Conversely speaking, the roughness parameter $k\sigma_\ell$ for the ice-bottom boundary is estimated to be 0.2 to 0.3.

This research mainly concentrated on the relation between the SAR observation data and the theoretical model output for the backscatter from lake ice. Even though the target of interest in this paper is on the lake ice in Northern Alaska, the methodology of this research can be used for more general targets, such as vegetation and sea ice. This research will contribute to the progress of understanding for microwave backscattering mechanisms.

ACKNOWLEDGMENTS

The main parts of this work were done during the author's stay in the University of Alaska Fairbanks as a visiting researcher for 1991 to 1992. This stay was made possible by NASDA and ASF.

The author is grateful to Professor Dr. W. F. Weeks and Dr. M. O. Jeffries of The University of Alaska Fairbanks for the valuable discussion and helpful suggestion on this work from a scientific view point. The author would also like to thank Mr. Kirk Walker and Ms. Kim Morris for the analysis of ice structure and stratigraphy.

It is also a great pleasure to thank Professor Mae of the Hokkaido University and Professor Nishio of the Hokkaido Educational University Kushiro for the continuous discussion on the ice physical property.

Reference

Bari, S. A.and J. Hallett, 1974. Nucleation and growth of bubbles at an ice–water interface, *Journal of Glaciology* ,13, pp. 489–520.

Bicknell, T, 1992. User's Guide To Products version 1.0, JPL Technical Note D-9362.

Cho, K., K.Takeda, K.Maeda and H.Wakabayashi, 1989. A study on sea ice monitoring using MOS-1/MSR, *Proceedings of International Geoscience and Remote Sensing Symposium*, pp. 991-994.

de Loor, G. P., 1956. Dielectric properties of heterogeneous mixtures containing water, *Journal of Microwave Power*, 3, pp. 67-73.

Elachi, C., M. L. Bryan and W. F. Weeks, 1976. Imaging radar observations of frozen arctic lakes, *Remote Sensing of Environment,* 5(3), pp. 169-175.

Fatland, D and A. Freeman, 1992. Calibration and change detection using Alaska SAR Facility ERS–1 SAR data. *Proceedings of International Geoscience and Remote Sensing Symposium*, 26-29 May, Houston TX, IEEE Cat. 92CH3041–1, 1164–1166.

Fujimoto, N., Y,Takahashi, T.Moriyama, M.Shimada, H.Wakabayashi, Y.Nakatani and S.Obayashi, 1989. Evaluation of SPOT HRV image data received in Japan, *Proceedings of International Geoscience and Remote Sensing Symposium*, pp. 463-466.

Gow, A. J. and D. Langston, 1977. Growth history of lake ice in relation to its stratigraphic, crystalline and mechanical structure. *CRREL Report 77–1*., Cold Regions Research and Engineering Laboratory, Hanover, New Hampshire.

Ishimaru, A., 1978. *Wave Propagation and Scattering in Random Media*, Vols. 1 and 2, Academic Press, New York.

Jeffries, M. O., H.Wakabayashi and W.F. Weeks, 1993. ERS-1 SAR backscatter changes associated with ice growing on shallow lakes in Arctic Alaska, *Proceedings of International Geoscience and Remote Sensing Symposium,* Volume IV, pp. 2001-2004.

Jeffries, M. O., K. Morris, W. F. Weeks and H. Wakabayashi, 1994. Structural and stratigraphic features and ERS–1 synthetic aperture radar backscatter characteristics of ice growing on shallow lakes in NW Alaska, winter 1991–92, *Journal of Geophysical Research,* Vol. 99, No. C11, pp. 22,459-22,471.

Maeda, K., H.Wakabayashi and H.Sato, 1987. Some results of MOS-1 airborne verification experiment-MSR (Microwave Scanning Radiometer), *Proceedings of International Geoscience and Remote Sensing Symposium*, pp.1073-1078.

Mellor, J., 1982. Bathymetry of Alaskan Arctic lakes: a key to resource inventory with remote sensing methods. Ph.D. Thesis, Institute of Marine Sciences, University of Alaska.

Rott, M. and C. Mätzler, 1987. Possibilities and limits of synthetic aperture radar for snow and glacier mapping. *Annals of Glaciology* 9, pp. 195–199.

Ruck, G. T., D.E.Barrick, W.D.Stuart, and C.K.Krickbaum, 1970. Radar Cross Section Handbook, vol. 1. New York : Prenum .

Sellmann, P. V., W. F. Weeks and W. J. Campbell, 1975. Use of side-looking airborne radar to determine lake depth on the Alaskan North Slope, *CRREL Special Report 230,* 6 pp., Cold Regions Research and Engineering Laboratory, Hanover, New Hampshire.

Shimada, M., 1995, Radiometric and geometric calibration of JERS-1 SAR, Advances in space research, Vol. 17, No. 1, pp. 79-88.

Ulaby, F. T., R. K. Moore and A. K. Fung, 1982. *Microwave Remote Sensing: Active and Passive, Volume II : Radar Remote Sensing and Surface Scattering and Emission Theory*, Artech House, Norwood, Massachusetts, 1982.

Ulaby, F. T., R. K. Moore and A. K. Fung, 1986. *Microwave Remote Sensing: Active and Passive, Volume III: Volume Scattering and Emission Theory, Advanced Systems, and Applications,* Addison-Wesley, Dedham, Massachusetts.

Wakabayashi, H., M. O. Jeffries and W. F. Weeks , 1992. C-band SAR backscatter from ice on shallow tundra lakes: observation and modelling, *Proceedings of the First ERS-1 Symposium–Space at the Service of our Environment*, Cannes France, European Space Agency Publication(ESA SP–359), pp. 333-337.

Wakabayashi, H, W. F. Weeks and M. O. Jeffries, 1993. A C–band backscatter model for lake ice in Alaska, *Proceedings of International Geoscience and Remote Sensing Symposium*,Volume III, pp. 1264-1266.

Wakabayashi, H., M. O. Jeffries and W.F. Weeks, 1994a. C-band backscatter variation and modelling for lake ice in northern Alaska, *Journal of The Remote Sensing Society of Japan*, Vol. 14, No. 3, pp. 18-27 (in Japanese).

Wakabayashi, H., F. Noshio, M. Aota and S. Takahashi, 1994b. A sea ice study using SAR data in the Sea of Okhotsk, *Journal of Geography*, Vol. 103, No. 6, pp. 674-683 (in Japanese).

Wakabayashi, H. and K. Arai, 1995a, A method of speckle noise reduction for SAR data, *Journal of the Japan Society of Photogrammetry and Remote Sensing*, Vol.34 , No. 2, pp. 15-24. (in Japanese)

Wakabayashi, H. , K. Hirano, F. Nishio, M. Aota and S. Takahashi, 1995b. A study of sea ice in the Sea of Okhotsk with SAR data", *Polar Record*, Vol. 31, No.178, pp.305-314.

Wakabayashi, H. and K. Arai , 1996a. A Method of Speckle Noise Reduction for SAR Data, *International Journal of Remote Sensing,* in press.

Wakabayashi, H. and K. Arai , 1996b. A New Method for SAR Speckle Noise Reduction (CST Filter), *Canadian Journal of Remote Sensing*, in press.

Weeks, W. F., A. G. Fountain, M. L. Bryan and C. Elachi, 1978. Differences in radar return from ice-covered North Slope lakes, *Journal of. Geophysical Research,* 83(C8), pp. 4069-4073.

Weeks, W. F., A. J. Gow and R. J. Schertler, 1981. Ground-truth observations of ice-covered North Slope lakes imaged by radar, *CRREL Research Report 81-19,* 17 pp., Cold Regions Research and Engineering Laboratory, Hanover, New Hampshire.

Weeks, W. F., P. Sellmann and W. J. Campbell, 1977. Interesting features of radar imagery of ice-covered North Slope lakes, *Journal of. Glaciology,* 18(78), pp. 129-136.

Appendix

Source codes for the lake ice backscattering model

Main program	93-111
Subroutine program	112-123

```
/*
        final_model_srm.c
                -1st. order sphere model
                -1st. order elipsoid model
                -1st. order cylinder model
                        Bubble is modeled as a sphere , a prolate spheroid or a cylinder
                -Bottom boundary is modelled as a small- perturbation

        Simulating Lake Ice (First order scattering)
                Calculating Backscatter coefficient of Lake Ice
                        1) created  by Hiro WAKABAYASHI on Nov.6,1992
                        2) modified on Apr.25,1993
                                -incorporate 3 models into one program
                        3) modified on May.12,1993
                                -fixed bug in ellipsoid model
                        4) modified on Aug.31,1993
                                -add surface roughness on the bottom boundary


*/

#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "nr.h"
#include "hiro_util.h"
#include "uni_nrutil.h"
#include "dcomplex.h"
#include "lakeice.h"


#define         Pai     3.1415926               /* Pai */
#define Degrad  0.0174533               /* Conversion coefficient degree to radian  */
#define Raddeg  57.29578                /* Conversion coefficient radian to degree  */
#define         Clv     299792458.0  /* Light Speed (m/sec) */


static int      nl;                             /* No. of layers in ice */

static dcomplex ptb,pab,pt[16],pa[16],pta[16];
                                                /* permittivity and permeability */

static float    freq,vi[16],a0[16],h0[16],d[16],ang[16],ka[16];
static int              bs[16];
                                                /* input parameter for each layer     */

static float    gm_bnd_h[16],gm_bub_h[16],k0_h[16],kp_h[16],

los_h[16],k0b_h[16],kpb_h[16],gm_bub_hd0[16],gm_bub_hdp[16];
static float    gm_bnd_v[16],gm_bub_v[16],k0_v[16],kp_v[16],

los_v[16],k0b_v[16],kpb_v[16],gm_bub_vd0[16],gm_bub_vdp[16];
                                                /* calculated parameters for each layer */

static float    ke0;
static dcomplex cea,ceb;

static float kl,kc,kf,t1,t2,kf0;
static dcomplex cvv1,cvv2,chh1,chh2,chv1;

static float    scl,shs;  /* Surface correlation length and height std. of bottom */

/*
*****************************************
*  Initialize input parameters     *
*****************************************
*/

void initParameter()
        {
        int i;
        ptb=Complex(1.0,0.0);
        pab=Complex(1.0,0.0);
        for (i=0;i<16;i++)
                {
```

```
                pt[i]=Complex(1.0,0.0);
                pa[i]=Complex(1.0,0.0);
                pta[i]=Complex(1.0,0.0);
                vi[i]=0.0; a0[i]=0.0; h0[i]=0.0 ; d[i]=0.0; ang[i]=0.0; ka[i]=0.0;
                bs[i]=1;
                gm_bnd_h[i]=0.0; gm_bub_h[i]=0.0; k0_h[i]=0.0; kp_h[i]=0.0; los_h[i]=0.0;
                gm_bnd_v[i]=0.0; gm_bub_v[i]=0.0; k0_v[i]=0.0; kp_v[i]=0.0; los_v[i]=0.0;
                k0b_h[i]=0.0; gm_bub_hd0[i]=0.0 ;k0b_v[i]=0.0; gm_bub_vd0[i]=0.0;
                kpb_h[i]=0.0; gm_bub_hdp[i]=0.0 ;kpb_v[i]=0.0; gm_bub_vdp[i]=0.0;
                scl=shs=0.0;
                }
        }
/*
*************************************
*  Input parameter routine          *
*************************************
*/
void inputParameter()
        {
        float  pr,pi,tv,ta,th,td;
        int      i,n,ib;

        printf("\nFrequency for calculation  (MHz) > ");
        scanf("%f",&freq);

        printf("\nPermittivity of upper media (air) (Real,Image) > " );
        scanf("%f,%f",&pr,&pi);
        pt[0]=Complex(pr,pi);

        printf("\nPermittivity of bottom media (water or soil) (Real,Image) > " );
        scanf("%f,%f",&pr,&pi);
        ptb=Complex(pr,pi);
        pab=Complex(1.0,0.0);

        printf("\nNumber of Layer in ice ( nl<=15) > " );
        scanf("%d",&nl);

        for (i=1;i<=nl;i++)
                {
                printf("\nPermittivity of ice layer %d (Real,Image) > ",i);
                scanf("%f,%f",&pr,&pi);
                pt[i]=Complex(pr,pi);
                printf("*****************  Layer     %d
*****************************",i);
                printf("\n*   vi:volume inc(percent)      a0:rad.of bubble(mm)  *");
                printf("\n*   h0:bubble length(cm)        d :ice thickness(cm)  *");
                printf("\n*    bs:bubble shape(0:sphere/1:ellipsoid/2:cylinder) *");

printf("\n*************************************************************");
                printf("\n  Layer parameter  (vi,a0,h0,d,bs)  > ");
                scanf("%f,%f,%f,%f,%d",&tv,&ta,&th,&td,&ib);
                vi[i]=tv*0.01; a0[i]=ta*0.001; h0[i]=th/2*0.01 ; d[i]=td*0.01; bs[i]=ib;
                }

        printf("\n*****  Bottom boundary surface roughness  *****" );
        printf("\nSurface cor. length(cm) & height std.(mm) (scl,shs) > " );
        scanf("%f,%f",&scl,&shs);
        scl=scl*0.01;
        shs=shs*0.001;

        }
/*
*************************************************************************
*  Calculate Dielectric constant                                        *
*    for 2 phase-mixture modelin the specific incident   *
*          -spheres or needles in host material       *
*          -needles are parallel to z-axis          *
*************************************************************************
(1)Title        :MixDE(p1t,p1a,p2th,p2ti,p2a,vif,th1,ibs)
(2)Input parameter
                p1t     (complex)       : permittivity of upper medium
                p1a     (complex)       : permeability of upper medium
```

```
                p2th(complex)  : permittivity of host
                p2ti(complex)  : permittivity of inclusion
                p2a     (complex)       : permeability for host and inclusion
                vif     (float)         : volume inclusion factor
                th1     (float)         : incident angle to the medium (degree)
                ibs (int)               : bubble shape  ( 0:sphere / 1:ellipsoid / 2:cylinder )
(3)Output parameter     : none
(4)Return
                MixDE (complex)         : permittivity of 2 phase-material

* /
dcomplex MixDE(p1t,p1a,p2th,p2ti,p2a,vif,th1,ibs)
dcomplex p1t,p1a,p2th,p2ti,p2a;
float    vif,th1;
int              ibs;
        {
        float a[3],b[2],btemp;
        dcomplex c[3],z;
        if (ibs==0) a[0]=a[1]=a[2]=1.0/3.0;
                else  { a[0]=a[1]=0.5; a[2]=0.0; }
        b[0]=0.0;
        b[1]=RefractAngle(p1t,p2th,p1a,p2a,th1);
        z=Mixture(p2th,p2ti,vif,a,b,c);
        btemp=RefractAngle(p1t,z,p1a,p2a,th1);
        while ( abs(btemp-b[1])>0.00001)                    /* derive refraction angle       */
                {                                                                       / *
iteration (threshold=0.00001) */
                b[1]=btemp;
                z=Mixture(p2th,p2ti,vif,a,b,c);
                btemp=RefractAngle(p1t,z,p1a,p2a,th1);
                }
        z=Mixture(p2th,p2ti,vif,a,b,c);
        return(z);
        }


/ *
**********************************
* Calculation for                 *
*       -average permittivity     *
*       -refract angle            *
*       -power reflectivity       *
*       -absorption coefficient   *
**********************************
(1)Title                : cal_para_ice

(p1t,p1a,p2th,p2ti,p2a,freq,vif,th1,ibs,th2,p2t_av,gam,ka)
(2)Input parameter
                p1t     (complex)       : permittivity of upper medium
                p1a (complex)  : permeability of upper medium
                p2th(complex)  : permittivity of host
                p2ta(complex)  : permittivity of inclusion
                p2a     (complex)       : permeability for host and inclusion
                freq(float)             : frequency (MHz)
                vif     (float)         : volume inclusion factor
                th1     (float)         : incident angle to the medium (degree)
                ibs (int)               : bubble shape  ( 0:sphere / 1:ellipsoid / 2:cylinder )

(2)Output parameter
                th2 (float)      : refract angle in the medium (degree)
                p2t_av(complex)         : average permittivity of the medium
                gam (float)      : power reflectivity of boundary
                                                gam[0] : for VV-Pol             gam[1] :
for HH-pol
                ka(float)               : volume absorption coefficient of the medium
* /

void  cal_para_ice(p1t,p1a,p2th,p2ti,p2a,freq,vif,th1,ibs,th2,p2t_av,gam,ka)
        dcomplex        p1t,p1a,p2th,p2ti,p2a,*p2t_av;
        float           freq,vif,th1;
        int                     ibs;
        float           *th2,gam[2],*ka;
        {
        dcomplex g1,g2,kai,z1;
        *p2t_av=MixDE(p1t,p1a,p2th,p2ti,p2a,vif,th1,ibs);
```

```
        z1=*p2t_av;
        *th2=RefractAngle(p1t,z1,p1a,p2a,th1);
        g1=ReflectVpol(p1t,z1,p1a,p2a,th1);
        gam[0]=Cabs(g1)*Cabs(g1);
        g2=ReflectHpol(p1t,z1,p1a,p2a,th1);
        gam[1]=Cabs(g2)*Cabs(g2);
        kai=AttPha(freq,z1);
        *ka=2.*kai.r;
        }


/*
************************
*    spm_btm_h     *
************************
        Calculate
                backscattering coefficient
                        of small perturbation model in H-polaritzation
        Surface is modeled as Gaussian correlation

(1)Title                :spm_btm_h
(2)Input parameter
        prmt1 (complex)         :Relative permittivity for layer 1
        prmt2 (complex)         :Relative permittivity for layer 2
        prma1 (complex) :Relative permeability for layer 1
        prma2 (complex) :Relative permeability for layer 2
        theta (int)             :Incident angle (degree)
        s       (float):Standard deviation of surface height
        l       (float):Surface correlation length (meter)
        freq  (float)   :Frequency      (MHz)
(3)Output parameter
        gam0  (float)   :Reflectivity in backscattering direction
        gamp  (float)   :Reflectivity in forwardscattering direction
(4)Return (float)       :None

*/

void spm_btm_h(prmt1,prmt2,prma1,prma2,theta,s,l,freq,gam0,gamp)
        dcomplex        prmt1,prmt2,prma1,prma2;
        float           theta,s,l,freq;
        float           *gam0,*gamp;
        {
        dcomplex c1;
        float     z1,z2,k,w;
        c1=ReflectHpol(prmt1,prmt2,prma1,prma2,theta);
        z1=Cabs(c1)*Cabs(c1);
        k=2*Pai/(Clv/(freq*1.0e6))*sqrt(prmt1.r);
        z2=0.5*l*l;
        w=exp((-1.0)*sqr((k*l*sin(Degrad*theta))));
        *gam0=8.0*pow(k,4.0)*(s*s)*pow(cos(Degrad*theta),4.0)*z1*z2*w/
                        (4.0*Pai*cos(Degrad*theta));
        *gamp=8.0*pow(k,4.0)*(s*s)*pow(cos(Degrad*theta),4.0)*z1*z2/
                        (4.0*Pai*cos(Degrad*theta));
        }

/*
************************
*    spm_btm_v     *
************************
        Calculate
                backscattering coefficient
                        of small perturbation model in V-polaritzation
        Surface is modeled as Gaussian correlation

(1)Title                :spm_btm_v
(2)Input parameter
        prmt1 (complex)         :Relative permittivity for layer 1
        prmt2 (complex)         :Relative permittivity for layer 2
        prma1 (complex) :Relative permeability for layer 1
        prma2 (complex) :Relative permeability for layer 2
        theta (int)             :Incident angle (degree)
        s       (float):Standard deviation of surface height
        l       (float):Surface correlation length (meter)
        freq  (float)   :Frequency      (MHz)
(3)Output parameter
```

```
        gam0 (float)   :Reflectivity in backscattering direction
        gamp (float)   :Reflectivity in forwardscattering direction
(4)Return (float)      :None

* /

void spm_btm_v(prmt1,prmt2,prma1,prma2,theta,s,l,freq,gam0,gamp)
        dcomplex        prmt1,prmt2,prma1,prma2;
        float           theta,s,l,freq;
        float           *gam0,*gamp;

        {
        dcomplex c1,c2,c3,c4,c5,c6,c7,c8,c9;
        float    z1,z2,k,w,sin2;
        sin2=sin(Degrad*theta)*sin(Degrad*theta);
        c1=Cdiv(prmt2,prmt1);
        c2.r=c1.r-1.0;
        c2.i=c1.i;
        c3=RCmul((1+sin2),c1);
        c3.r=sin2-c3.r;
        c3.i=(-1.0)*c3.i;
        c4=Cmul(c2,c3);
        c5=RCmul(cos(Degrad*theta),c1);
        c6.r=c1.r-sin2;
        c6.i=c1.i;
        c7=Cadd(c5,Csqrt(c6));
        c8=Cmul(c7,c7);
        c9=Cdiv(c4,c8);
        z1=Cabs(c9)*Cabs(c9);
        k=2*Pai/(Clv/(freq*1.0e6))*sqrt(prmt1.r);
        z2=0.5*l*l;
        w=exp((-1.0)*sqr((k*l*sin(Degrad*theta))));
        *gam0=8.0*pow(k,4.0)*(s*s)*pow(cos(Degrad*theta),4.0)*z1*z2*w/
                (4.0*Pai*cos(Degrad*theta));
        *gamp=8.0*pow(k,4.0)*(s*s)*pow(cos(Degrad*theta),4.0)*z1*z2/
                (4.0*Pai*cos(Degrad*theta));
        }

/ *
**************************************************************
*       Calculate Bubble Scatter                              *
*               - Volume scattering coefficient         *
*               - Volume scattering loss                      *
*                       consider bubble as "sphere"     *
**************************************************************
Calculate volume scattering coefficent for bubble
(1) Title                               :cal_bubble_sph  (er,mr,a0,h0,vi,freq,in_ang,ks)
(2) Input Parameter
                        er (dcomplex)  :relative permittivity of host to bubble
                        mr (dcomplex)  :relative parmiability of host to bubble
                        a0 (float)             :radius of bubble (meter)
                        h0 (float)             :half length of bubble (meter)
                        vi (float)             :volume inclusion factor
                        freq (float)     :frequency (MHz)
                        in_ang(float)    :incident angle (degree)
(3) Output Parametar
                        ks[] (float)     :volume scattering coefficient
                        ks[0]                   :0 degree vv    (specular scattering)
                        ks[1]                   :180 degree vv (scattering extinction)
                        ks[2]                   :0 degree hh    (specular scattering)
                        ks[3]                   :180 degree hh (scattering extinction)
                        ks[4]                   :0 degree vv    (direct backscattering)
                        ks[5]                   :180 degree vv (direct backscattering)
                        ks[6]                   :0 degree hh    (direct backscattering)
                        ks[7]                   :180 degree hh (direct backscattering)

(4) Return                              :none
* /

void cal_bubble_sph(er,mr,a0,h0,vi,freq,in_ang,ks)
dcomplex        er,mr;
float           a0,h0,vi,freq,in_ang;
float           ks[8];
        {
        dcomplex fer,fmr,ca1,ca2,cb1,cb2,cr,ci,ctmp1,ctmp2,m1;
```

```
float    k0,k1,k1a,sig_ex,ftmp,c1ab,c2ab,nn;
float pusi;
int      i;

cr=Complex(1.0,0.0);
ci=Complex(0.0,1.0);
fer=Cdiv(cr,er);
fmr=Cdiv(cr,mr);
m1=fer;
k0=2.0*Pai/(Clv/(freq*1.0e6));
k1=k0*sqrt(er.r);
k1a=k1*a0;
ca1=Cmul(ci,RCmul(pow(k1a,5.0)/45.0,Csub(m1,cr)));
ca2=Complex(0.0,0.0);
ctmp1=RCmul(pow(k1a,3.0),Cdiv(Csub(m1,cr),Cadd(m1,RCmul(2.0,cr))));
ctmp2=Cmul(Csub(m1,cr),Csub(m1,RCmul(2.0,cr)));
ctmp2=Cdiv(ctmp2,Cmul(Cadd(m1,RCmul(2.0,cr)),Cadd(m1,RCmul(2.0,cr))));
ctmp2=RCmul(3.0/5.0*pow(k1a,5.0),ctmp2);
ctmp1=Cadd(ctmp1,ctmp2);
cb1=Cmul(RCmul((-2.0)/3.0,ci),ctmp1);
ctmp1=Cdiv(Csub(m1,cr),Cadd(RCmul(2.0,m1),RCmul(3.0,cr)));
cb2=Cmul(RCmul((-1.0)*pow(k1a,5.0)/15.0,ci),ctmp1);
c1ab=Cabs(Cmul(ca1,ca1))+Cabs(Cmul(cb1,cb1));
c2ab=Cabs(Cmul(ca2,ca2))+Cabs(Cmul(cb2,cb2));
sig_ex=2.0*Pai/pow(k1,2.0)*(3.0*c1ab+5.0*c2ab);
ftmp=4.0*Pai*a0*a0*pow(k1a,4.0)*
                pow(Cabs(Cdiv(Csub(m1,cr),Cadd(m1,RCmul(2.0,cr)))),2.0);

pusi=Degrad*(90-in_ang);
ks[0]=ks[2]=ftmp*cos(2.0*pusi)*cos(2.0*pusi)/(4.0*Pai);
ks[4]=ks[6]=ftmp/(4.0*Pai);
ks[5]=ks[7]=ftmp*cos(2.0*(Pai/2.0-pusi))*cos(2.0*(Pai/2.0-pusi))/(4.0*Pai);
ks[1]=sig_ex;
ks[3]=sig_ex;

nn=vi/(4.0/3.0*Pai*a0*a0*h0);
for (i=0;i<=7;i++)
        ks[i]=nn*ks[i];
}


/ *

**************************************************************
*       Calculate Bubble Scatter                              *
*               - Volume scattering coefficient         *
*               - Volume scattering loss                      *
*                       consider bubble as "prolate ellipsoid"*
**************************************************************
Calculate volume scattering coefficent for bubble
(1) Title                                       :cal_bubble_elp  (er,mr,a0,h0,vi,freq,in_ang,ks)
(2) Input Parameter
                er (dcomplex)   :relative permittivity of host to bubble
                mr (dcomplex)   :relative parmiability of host to bubble
                a0 (float)              :radius of bubble (meter)
                h0 (float)              :half length of bubble (meter)
                vi (float)              :volume inclusion factor
                freq (float)    :frequency (MHz)
                in_ang(float)   :incident angle (degree)
(3) Output Parametar
                ks[] (float)    :volume scattering coefficient
                ks[0]                   :0 degree vv    (specular scattering)
                ks[1]                   :180 degree vv (scattering extinction)
                ks[2]                   :0 degree hh    (specular scattering)
                ks[3]                   :180 degree hh (scattering extinction)
                ks[4]                   :0 degree vv    (direct backscattering)
                ks[5]                   :180 degree vv (direct backscattering)
                ks[6]                   :0 degree hh    (direct backscattering)
                ks[7]                   :180 degree hh (direct backscattering)

(4) Return                              :none
* /

float ke1(er,freq)      /* (16 * Pai / 9 * k^4) */
dcomplex er;
```

```
float freq;
        {
        float k0,c;
        k0=2.0*Pai/(Clv/(freq*1.0e6));
        c=k0*sqrt(er.r);
        c=16.0*Pai/9.0*pow(c,4.0);
        return c;
        }

float eesa(x)                           /* equivalent ellipsoid semi-axis a & b */
float x;
        {
        float c;
        c=pow(3./2.,1./3.);
        c=c*x;
        return c;
        }

float iae(ae,be)                /* Ia  (elliptic integral a ) */
float ae,be;
        {
        float c,c1,c2,c3;
        c1=ae/be;
        c2=c1*c1-1.0;
        c3=ae*be*be;
        c=(2.0/c3)*(1.0/c2)*(c1/(2.0*sqrt(c2))*log((c1+sqrt(c2))/(c1-sqrt(c2)))-1.0);
        return c;
        }

float ibe(ae,be)                /* Ib  (elliptic integral b) */
float ae,be;
        {
        float c,c1,c2,c3;
        c1=ae/be;
        c2=c1*c1-1.0;
        c3=ae*be*be;
        c=(2.0/c3)*(c1/(2.0*c2))*(c1-1./(2.0*sqrt(c2))*log((c1+sqrt(c2))/(c1-sqrt(c2))));
        return c;
        }

dcomplex fia(er,a0,h0)          /* (er-1)/((er-1)*Ia+2./(a*b^2)) */
dcomplex er;
float a0,h0;
        {
        dcomplex z,z1;
        float a1,b1,ia,c;
        z1=Csub(er,Complex(1.0,0.0));
        a1=eesa(h0); b1=eesa(a0);
        ia=iae(a1,b1);
        c=2.0/(a1*b1*b1);
        z=Cdiv(z1,Cadd(RCmul(ia,z1),Complex(c,0.0)));
        return z;
        }

dcomplex fib(er,a0,h0)          /* (er-1)/((er-1)*Ib+2./(a*b^2)) */
dcomplex er;
float a0,h0;
        {
        dcomplex z,z1;
        float a1,b1,ib,c;
        z1=Csub(er,Complex(1.0,0.0));
        a1=eesa(h0); b1=eesa(a0);
        ib=ibe(a1,b1);
        c=2.0/(a1*b1*b1);
        z=Cdiv(z1,Cadd(RCmul(ib,z1),Complex(c,0.0)));
        return z;
        }

float fuu(x,y)                          /* 1.0-sin(x)^2*cos(y)^2      * /
float x,y;
        {
        float c;
        c=1.0-sin(x)*sin(x)*cos(y)*cos(y);
        return c;
        }
```

```
float cs_pe_hh(phi,pui,pus)             /* Radar cross section for hh-pol ( one particle ) */
float phi,pui,pus;                              /* pui : dummy parameter */
        {
        float c;
        c=ke0*Cabs(Cmul(ceb,ceb))*fuu(phi,pus);
        return c;
        }

float cs_pe_vv(phi,pui,pus)             /* Radar cross section for vv-pol ( one particle ) */
float phi,pui,pus;
        {
        dcomplex z1,z2,z3;
        float c1,c2,c3,c;
        c1=(cos(pui)*cos(pus)*cos(phi))*(cos(pui)*cos(pus)*cos(phi))/fuu(phi,pus);
        z1=RCmul(c1,Cmul(cea,cea));
        c2=(sin(pui)*sin(pus))*(sin(pui)*sin(pus))/fuu(phi,pus);
        z2=RCmul(c2,Cmul(ceb,ceb));
        c3=2.0*(cos(pui)*cos(pus)*cos(phi))*(sin(pui)*sin(pus))/fuu(phi,pus);
        z3=RCmul(c3,Cmul(cea,ceb));
        c=ke0*Cabs(Cadd(Cadd(z1,z2),z3));
        return c;
        }

float cs_pe_hv(phi,pui,pus)             /* Radar cross section for hv-pol ( one particle ) */
float phi,pui,pus;
        {
        dcomplex z1,z2,z3;
        float c1,c2,c3,c;
        c1=(cos(phi)*sin(phi)*sin(pui)*cos(pus)*cos(pus))*
                        (cos(phi)*sin(phi)*sin(pui)*cos(pus)*cos(pus))/fuu(phi,pus);
        z1=RCmul(c1,Cmul(ceb,ceb));
        c2=(sin(phi)*cos(pui)*sin(pus)*cos(pus))*
                        (sin(phi)*cos(pui)*sin(pus)*cos(pus))/fuu(phi,pus);
        z2=RCmul(c2,Cmul(cea,cea));
        c3=(-2.0)*(cos(phi)*sin(phi)*sin(pui)*cos(pus)*cos(pus))*
                        (sin(phi)*cos(pui)*sin(pus)*cos(pus))/fuu(phi,pus);
        z3=RCmul(c3,Cmul(cea,ceb));
        c=ke0*Cabs(Cadd(Cadd(z1,z2),z3));
        return c;
        }

void cal_para_elp(a0,h0,er,mr,freq)
float   a0,h0,freq;
dcomplex er,mr;
        {
        dcomplex fer,fmr;
        fer=Cdiv(Complex(1.0,0.0),er);
        fmr=Cdiv(Complex(1.0,0.0),mr);
        ke0=ke1(er,freq);
        cea=fia(fer,a0,h0);
        ceb=fib(fer,a0,h0);
        }

void cal_bubble_elp(er,mr,a0,h0,vi,freq,in_ang,ks)
dcomplex        er,mr;
float           a0,h0,vi,freq,in_ang;
float           ks[8];
        {
        float pusi;
        float qvv,qhh,qhv,qvh,qfc,nn,c;
        dcomplex z1,z2;
        int     i;
        pusi=Degrad*(90-in_ang);
        cal_para_elp(a0,h0,er,mr,freq);
        ks[0]=cs_pe_vv(0.0,pusi,-1.*pusi)/(4.0*Pai);
        ks[2]=cs_pe_hh(0.0,pusi,-1.*pusi)/(4.0*Pai);
        ks[4]=cs_pe_vv(0.0,pusi,pusi)/(4.0*Pai);
        ks[6]=cs_pe_hh(0.0,pusi,pusi)/(4.0*Pai);
        ks[5]=cs_pe_vv(Pai,pusi,pusi)/(4.0*Pai);
        ks[7]=cs_pe_hh(Pai,pusi,pusi)/(4.0*Pai);
        qhh=ke0*Cabs(Cmul(ceb,ceb))*8.37758;
        qvv=ke0*Cabs(Cmul(cea,cea))*cos(pusi)*cos(pusi)*12.625;
        qhv=0.0;
        z1=RCmul(sin(pusi)*sin(pusi)*8.375,Cmul(ceb,ceb)),
```

```
        z2=RCmul((-2.0)*cos(pusi)*sin(pusi)*(-4.25),Cmul(ceb,cea));
        c=Cabs(Cadd(z1,z2));
        qvh=ke0*c;
        ks[1]=(qvv+qvh)/(4.0*Pai);
        ks[3]=(qhh+qhv)/(4.0*Pai);
        nn=vi/(Pai*a0*a0*h0*2.0);
        for (i=0;i<=7;i++)
                ks[i]=nn*ks[i];
        }

/*

**************************************************************
*       Calculate Bubble Scatter                              *
*               - Volume scattering coefficient :kais_0   *
*               - Volume scattering loss                :kais_p   *
*                       consider bubble as finite cylinder    *
**************************************************************
Calculate volume scattering coefficent for bubble
(1) Title                                       :cal_bubble_cyl  (er,mr,a0,h0,vi,freq,in_ang,ks)
(2) Input Parameter
                        er (dcomplex)   :relative permittivity of host to bubble
                        mr (dcomplex)  :relative parmiability of host to bubble
                        a0 (float)              :radius of bubble (meter)
                        h0 (float)              :length of bubble (meter)
                        vi (float)              :volume inclusion factor
                        freq (float)      :frequency (MHz)
                        in_ang(float)     :incident angle (degree)
(3) Output Parametar
                        ks[] (float)      :volume scattering coefficient
                        ks[0]                   :0 degree vv    (specular scattering)
                        ks[1]                   :180 degree vv (scattering extinction)
                        ks[2]                   :0 degree hh    (specular scattering)
                        ks[3]                   :180 degree hh (scattering extinction)
                        ks[4]                   :0 degree vv    (direct backscattering)
                        ks[5]                   :180 degree vv (direct backscattering)
                        ks[6]                   :0 degree hh    (direct backscattering)
                        ks[7]                   :180 degree hh (direct backscattering)

(4) Return                                      :none
* /


double ke(er,freq)                              /* wave number in host material */
dcomplex er;
float freq;
        {
        double k0,c;
        k0=2.0*Pai/(Clv/(freq*1.0e6));
        c=k0*sqrt(er.r);
        return c;
        }

double c1(a0,er,pusaii,freq)    /* Pai^2*a0/(4.0*cos(pi)^2)*(k*a0)^3 */
dcomplex        er;
float   a0,pusaii,freq;
        {
        double  c1,c;
        c1=Pai*Pai*a0/(4.0*cos(pusaii)*cos(pusaii));
        c=c1*power2((ke(er,freq)*a0),3);
        return c;
        }

dcomplex c2(er,pusaii)                  /* (er-1)*cos(pi)^2    * /
dcomplex        er;
float   pusaii;
        {
        dcomplex z1,z;
        z1=Csub(er,Complex(1.0,0.0));
        z=RCmul((cos(pusaii)*cos(pusaii)),z1);
        return z;
        }

dcomplex c3(er,mr,pusaii)               /*  2.0*((er+1)*(mr-1)+sin(pi)^2*(er-
1)*(mr+1))/((er+1)*(mr+1)) * /
```

```
dcomplex      er,mr;
float   pusaii;
        {
        dcomplex z1,z2,z3,z;
        z1=Cmul(Cadd(er,Complex(1.0,0.0)),Csub(mr,Complex(1.0,0.0)));
        z2=Cmul(Csub(er,Complex(1.0,0.0)),Cadd(mr,Complex(1.0,0.0)));
        z3=Cmul(Cadd(er,Complex(1.0,0.0)),Cadd(mr,Complex(1.0,0.0)));
        z2=RCmul((sin(pusaii)*sin(pusaii)),z2);
        z1=Cadd(z1,z2);
        z1=Cdiv(z1,z3);
        z=RCmul(2.0,z1);
        return z;
        }

dcomplex c4(mr,pusaii)                  /* (mr-1)*cos(pi)^2   * /
dcomplex      mr;
float   pusaii;
        {
        dcomplex z1,z;
        z1=Csub(mr,Complex(1.0,0.0));
        z=RCmul((cos(pusaii)*cos(pusaii)),z1);
        return z;
        }


dcomplex c5(er,mr,pusaii)               /*  2.0*((er-1)*(mr+1)+sin(pi)^2*(er+1)*(mr-
1))/((er+1)*(mr+1))   * /
dcomplex      er,mr;
float   pusaii;
        {
        dcomplex z1,z2,z3,z;
        z1=Cmul(Csub(er,Complex(1.0,0.0)),Cadd(mr,Complex(1.0,0.0)));
        z2=Cmul(Cadd(er,Complex(1.0,0.0)),Csub(mr,Complex(1.0,0.0)));
        z3=Cmul(Cadd(er,Complex(1.0,0.0)),Cadd(mr,Complex(1.0,0.0)));
        z2=RCmul((sin(pusaii)*sin(pusaii)),z2);
        z1=Cadd(z1,z2);
        z1=Cdiv(z1,z3);
        z=RCmul(2.0,z1);
        return z;
        }


double c6(a0,er,pusaii,freq)    /* 4.0*Pai^2*a0/(cos(pi)^2)*(k*a0)^3 */
dcomplex      er;
float   a0,pusaii,freq;
        {
        float   c1,c;
        c1=4.0*Pai*Pai*a0/(cos(pusaii)*cos(pusaii));
        c=c1*power2((ke(er,freq)*a0),3);
        return c;
        }

dcomplex c7(er,mr,pusaii)               /* (mr*er-1)/((er+1)*(mr+1))*sin(pi)   */
dcomplex      er,mr;
float   pusaii;
        {
        dcomplex z1,z2,z;
        z1=Csub(Cmul(er,mr),Complex(1.0,0.0));
        z2=Cmul(Cadd(er,Complex(1.0,0.0)),Cadd(mr,Complex(1.0,0.0)));
        z1=RCmul(sin(pusaii),z1);
        z=Cdiv(z1,z2);
        return z;
        }

double c8(h0,er,freq)                   /* 4.0*k*h^2/Pai   */
dcomplex      er;
float   h0,freq;
        {
        float   c;
        c=4.0*ke(er,freq)*h0*h0/Pai;
        return c;
        }

float cs_ic_vv(x)                       /* Cross section for infinite cylinder-vv */
```

```
float x;
        {
        dcomplex z1;
        float c;
        z1=Csub(cvv1,RCmul(cos(x),cvv2));
        c=kl*Cabs(z1)*Cabs(z1);
        return c;
        }

float cs_ic_hh(x)                       /* Cross section for infinite cylinder-hh */
float x;
        {
        dcomplex z1;
        float c;
        z1=Csub(chh1,RCmul(cos(x),chh2));
        c=kl*Cabs(z1)*Cabs(z1);
        return c;
        }

float cs_ic_hv(x)                       /* Cross section for infinite cylinder-hv,vh */
float x;
        {
        dcomplex z1;
        float c;
        z1=RCmul(sin(x),chv1);
        c=kc*Cabs(z1)*Cabs(z1);
        return c;
        }

float cs_fc(x)                          /* Cross section for finite cylinder */
float x;
        {
        float c1,c2,c;
        c1=cos(x)*cos(x)*kf;
        c2=t1*(t2+sin(x));
        c=c1*power2(sin(c2)/c2,2);
        return c;
        }

float cs_fc_i(x)                        /* Cross section for finite cylinder for integration */
float x;
        {
        float c1,c2,c;
        c1=cos(x)*cos(x)*kf;
        c2=t1*(t2+sin(x));
        c=c1*power2(sin(c2)/c2,2)*cos(x);
        return c;
        }


void  cal_para_cyl(a0,h0,pusi,er,mr,freq)
float    a0,h0,pusi,freq;
dcomplex  er,mr;
        {
        dcomplex  fer,fmr;
        kl=c1(a0,er,pusi,freq);
        fer=Cdiv(Complex(1.0,0.0),er);
        fmr=Cdiv(Complex(1.0,0.0),mr);
        cvv1=c2(fer,pusi);
        cvv2=c3(fer,fmr,pusi);
        chh1=c4(fmr,pusi);
        chh2=c5(fer,fmr,pusi);
        kc=c6(a0,er,pusi,freq);
        chv1=c7(fer,fmr,pusi);
        kf=c8(h0,er,freq);
        t1=ke(er,freq)*h0;
        t2=sin(pusi);
        kf0=kf*power2(cos(pusi),2);
        }


void  cal_bubble_cyl(er,mr,a0,h0,vi,freq,in_ang,ks)
dcomplex        er,mr;
float           a0,h0,vi,freq,in_ang;
float           ks[8];
```

```
{
float pusi;
float qvv,qhh,qhv,qfc,nn;
int     i;
pusi=Degrad*(90-in_ang);
cal_para_cyl(a0,h0,pusi,er,mr,freq);
ks[0]=cs_ic_vv(0.0)*kf0/(4.0*Pai);
ks[2]=cs_ic_hh(0.0)*kf0/(4.0*Pai);
ks[4]=cs_ic_vv(0.0)*cs_fc(pusi)/(4.0*Pai);
ks[5]=cs_ic_vv(Pai)*cs_fc(pusi)/(4.0*Pai);
ks[6]=cs_ic_hh(0.0)*cs_fc(pusi)/(4.0*Pai);
ks[7]=cs_ic_hh(Pai)*cs_fc(pusi)/(4.0*Pai);
qvv=qromb(cs_ic_vv,0.0,2.0*Pai);          /* qromb : cal romberg integration */
qhh=qromb(cs_ic_hh,0.0,2.0*Pai);
qhv=qromb(cs_ic_hv,0.0,2.0*Pai);
qfc=qromb(cs_fc_i,(-0.5*Pai),0.5*Pai);
qvv=qvv*qfc/(4.0*Pai);
qhh=qhh*qfc/(4.0*Pai);
qhv=qhv*qfc/(4.0*Pai);
ks[1]=qvv+qhv;
ks[3]=qhh+qhv;
nn=vi/(Pai*a0*a0*h0*2.0);
for (i=0;i<=7;i++)
        ks[i]=nn*ks[i];
}


/*
*****************************************
        Calculation of
                -refraction angle
                -average permittivity
                -volume absorption coefficient
                -volume scattering coefficient
                -power loss factor
                -power reflectivity

                for each ice layer

*****************************************
(1)Title        :cal_layer(theta)
(2)Input parameter
                theta(float)     : incident angle to ice (degree)
(3)Output parameter    : none
(4)Return                               : none
*/

void cal_layer(th1)
float th1;
        {
        int i;
        dcomplex pt_bub,pt_av,gamma;
        float  th,gam[2],kai,ks[8],zc;
        pt_bub=Complex(1.0,0.0);
        ang[0]=th1;
        for (i=1;i<=nl;i++)
                {
                cal_para_ice(pt[i-1],pa[i-1],pt[i],pt_bub,pa[i],
                                                freq,vi[i],ang[i-1],bs[i],&th,&pt_av,gam,&kai);
                ang[i]=th;
                pta[i]=pt_av;
                ka[i]=kai;
                gm_bnd_v[i-1]=gam[0];
                gm_bnd_h[i-1]=gam[1];

                if (bs[i]==0)
                        cal_bubble_sph(pt[i],pa[i],a0[i],h0[i],vi[i],freq,ang[i],ks);
                else if (bs[i]==1)
                        cal_bubble_elp(pt[i],pa[i],a0[i],h0[i],vi[i],freq,ang[i],ks);
                else if (bs[i]==2)
                        cal_bubble_cyl(pt[i],pa[i],a0[i],h0[i],vi[i],freq,ang[i],ks);

                k0_v[i]=ks[0]; k0_h[i]=ks[2]; k0b_v[i]=ks[4]; k0b_h[i]=ks[6];
                kp_v[i]=ks[1]; kp_h[i]=ks[3]; kpb_v[i]=ks[5]; kpb_h[i]=ks[7];
```

```
                los_v[i]=exp((kp_v[i]+ka[i])*d[i]/cos(Degrad*ang[i]));
                los_h[i]=exp((kp_h[i]+ka[i])*d[i]/cos(Degrad*ang[i]));
                gm_bub_v[i]=k0_v[i]*d[i]/cos(Degrad*ang[i]);
                gm_bub_h[i]=k0_h[i]*d[i]/cos(Degrad*ang[i]);
                gm_bub_vd0[i]=k0b_v[i]/(2*(kp_v[i]+ka[i]))*(1.0-1/(los_v[i]*los_v[i]));
                gm_bub_hd0[i]=k0b_h[i]/(2*(kp_h[i]+ka[i]))*(1.0-1/(los_h[i]*los_h[i]));
                gm_bub_vdp[i]=kpb_v[i]/(2*(kp_v[i]+ka[i]))*(1.0-1/(los_v[i]*los_v[i]));
                gm_bub_hdp[i]=kpb_h[i]/(2*(kp_h[i]+ka[i]))*(1.0-1/(los_h[i]*los_h[i]));
                }
        cal_para_ice(pt[nl],pa[nl],ptb,pt_bub,pab,
                                        freq,0.0,ang[nl],1,&th,&pt_av,gam,&kai);
        gm_bnd_v[nl]=gam[0];
        gm_bnd_h[nl]=gam[1];
        }

/ *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
        calculate
        cascade matrix of v-pol
                        for layer l

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
(1)Title        :cas_v(l)
(2)Input parameter
                l(int)   :layer no
(3)Output parameter    : none
(4)Return                               : cascade matrix ( 4 x 4 )
* /

float **cas_v(l)
int l;
        {
        float   **ma,**mb,**mc,**md,**mt,**mai;
        float **cma,det;
        int i,j;
        ma=matrix(1,2,1,2);  mb=matrix(1,2,1,2);
        mc=matrix(1,2,1,2);  md=matrix(1,2,1,2);
        mai=matrix(1,2,1,2);  mt=matrix(1,2,1,2);
        cma=matrix(1,4,1,4);
        ma[1][1]=ma[2][2]=(1.-gm_bnd_v[l-1])/los_v[l];
        ma[1][2]=ma[2][1]=(1.-gm_bnd_v[l-1])*gm_bub_v[l]/los_v[l];
        mb[1][1]=mb[2][2]=(1.-gm_bnd_v[l])*gm_bnd_v[l-1]/(los_v[l]*los_v[l])+
                                        (1.-gm_bnd_v[l])*gm_bub_vdp[l];
        mb[1][2]=mb[2][1]=2.*(1.-gm_bnd_v[l])*gm_bnd_v[l-
1]*gm_bub_v[l]/(los_v[l]*los_v[l])+
                                        (1.-gm_bnd_v[l])*gm_bub_vd0[l];
        mc[1][1]=mc[2][2]=(1.-gm_bnd_v[l-1])*gm_bnd_v[l]/(los_v[l]*los_v[l])+
                                        (1.-gm_bnd_v[l-1])*gm_bub_vdp[l];
        mc[1][2]=mc[2][1]=2.*(1.-gm_bnd_v[l-
1])*gm_bnd_v[l]*gm_bub_v[l]/(los_v[l]*los_v[l])+
                                        (1.-gm_bnd_v[l-1])*gm_bub_vd0[l];
        md[1][1]=md[2][2]=(1.-gm_bnd_v[l])/los_v[l];
        md[1][2]=md[2][1]=(1.-gm_bnd_v[l])*gm_bub_v[l]/los_v[l];
        det=ma[1][1]*ma[2][2]-ma[1][2]*ma[2][1];
        mai[1][1]=ma[2][2]/det;
        mai[1][2]=-1.*ma[1][2]/det;
        mai[2][1]=-1.*ma[2][1]/det;
        mai[2][2]=ma[1][1]/det;
        for (i=1;i<=2;i++)
                for (j=1;j<=2;j++)
                        cma[i][j]=mai[i][j];
        mt=mul_matrices(mai,mb,2);
        for (i=1;i<=2;i++)
                for (j=3;j<=4;j++)
                        cma[i][j]=-1.*mt[i][j-2];
        mt=mul_matrices(mc,mai,2);
        for (i=3;i<=4;i++)
                for (j=1;j<=2;j++)
                        cma[i][j]=mt[i-2][j];
        ma=mul_matrices(mc,mai,2);
        mc=mul_matrices(ma,mb,2);
        mt=sub_matrices(md,mc,2);
        for (i=3;i<=4;i++)
                for (j=3;j<=4;j++)
                        cma[i][j]=mt[i-2][j-2];
```

```
        return cma;
        free_matrix(ma,1,2,1,2);
        free_matrix(mb,1,2,1,2);
        free_matrix(mc,1,2,1,2);
        free_matrix(md,1,2,1,2);
        free_matrix(mai,1,2,1,2);
        free_matrix(mt,1,2,1,2);
        free_matrix(cma,1,4,1,4);
        }

/ *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
        calculate
        cascade matrix of v-pol
                        for layer l

        incorporate small purtubation model
                into the bottom boundary

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
(1)Title          :cas_v_srm(l)
(2)Input parameter
                l(int)   :layer no
(3)Output parameter    : none
(4)Return                                   : cascade matrix ( 4 x 4 )
* /

float  **cas_v_srm(l)
int l;
        {
        float    **ma,**mb,**mc,**md,**mt,**mai;
        float    **cma,det;
        float    gm_btm_0,gm_btm_p;
        int  i,j;

        ma=matrix(1,2,1,2);  mb=matrix(1,2,1,2);
        mc=matrix(1,2,1,2);  md=matrix(1,2,1,2);
        mai=matrix(1,2,1,2);  mt=matrix(1,2,1,2);
        cma=matrix(1,4,1,4);

        spm_btm_v(pta[nl],ptb,pa[nl],pab,ang[nl],shs,scl,freq,&gm_btm_0,&gm_btm_p);

        ma[1][1]=ma[2][2]=(1.-gm_bnd_v[l-1])/los_v[l];
        ma[1][2]=ma[2][1]=(1.-gm_bnd_v[l-1])*gm_bub_v[l]/los_v[l];
        mb[1][1]=mb[2][2]=(1.-gm_bnd_v[l])*gm_bnd_v[l-1]/(los_v[l]*los_v[l])+
                                        (1.-gm_bnd_v[l])*gm_bub_vdp[l];
        mb[1][2]=mb[2][1]=2.*(1.-gm_bnd_v[l])*gm_bnd_v[l-
1]*gm_bub_v[l]/(los_v[l]*los_v[l])+
                                        (1.-gm_bnd_v[l])*gm_bub_vd0[l];
        mc[1][1]=mc[2][2]=(1.-gm_bnd_v[l-1])*gm_bnd_v[l]/(los_v[l]*los_v[l])+
                                        (1.-gm_bnd_v[l-1])*gm_bub_vdp[l];
        mc[1][2]=mc[2][1]=2.*(1.-gm_bnd_v[l-
1])*gm_bnd_v[l]*gm_bub_v[l]/(los_v[l]*los_v[l])+
                                        (1.-gm_bnd_v[l-1])*gm_bub_vd0[l]+
                                        (1.-gm_bnd_v[l-
1])*gm_btm_0/(los_v[l]*los_v[l]);
        md[1][1]=md[2][2]=(1.-gm_bnd_v[l])/los_v[l];
        md[1][2]=md[2][1]=(1.-gm_bnd_v[l])*gm_bub_v[l]/los_v[l];
        det=ma[1][1]*ma[2][2]-ma[1][2]*ma[2][1];
        mai[1][1]=ma[2][2]/det;
        mai[1][2]=-1.*ma[1][2]/det;
        mai[2][1]=-1.*ma[2][1]/det;
        mai[2][2]=ma[1][1]/det;
        for (i=1;i<=2;i++)
                for (j=1;j<=2;j++)
                        cma[i][j]=mai[i][j];
        mt=mul_matrices(mai,mb,2);
        for (i=1;i<=2;i++)
                for (j=3;j<=4;j++)
                        cma[i][j]=-1.*mt[i][j-2];
        mt=mul_matrices(mc,mai,2);
        for (i=3;i<=4;i++)
                for (j=1;j<=2;j++)
                        cma[i][j]=mt[i-2][j];
        ma=mul_matrices(mc,mai,2);
```

```
        mc=mul_matrices(ma,mb,2);
        mt=sub_matrices(md,mc,2);
        for (i=3;i<=4;i++)
                for (j=3;j<=4;j++)
                        cma[i][j]=mt[i-2][j-2];
        return cma;
        free_matrix(ma,1,2,1,2);
        free_matrix(mb,1,2,1,2);
        free_matrix(mc,1,2,1,2);
        free_matrix(md,1,2,1,2);
        free_matrix(mai,1,2,1,2);
        free_matrix(mt,1,2,1,2);
        free_matrix(cma,1,4,1,4);
        }

/ *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
        calculate
        cascade matrix of h-pol
                        for layer l

* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
(1)Title        :cas_h(l)
(2)Input parameter
                l(int)   :layer no
(3)Output parameter    : none
(4)Return                               : cascade matrix ( 4 x 4 )
* /

float **cas_h(l)
int l;
        {
        float   **ma,**mb,**mc,**md,**mt,**mai;
        float **cma,x,det;
        int i,j;
        ma=matrix(1,2,1,2);  mb=matrix(1,2,1,2);
        mc=matrix(1,2,1,2);  md=matrix(1,2,1,2);
        mai=matrix(1,2,1,2);  mt=matrix(1,2,1,2);
        cma=matrix(1,4,1,4);
        x=gm_bnd_h[l-1]*gm_bnd_h[l]/(los_h[l]*los_h[l]);
        ma[1][1]=ma[2][2]=(1.-gm_bnd_h[l-1])/los_h[l];
        ma[1][2]=ma[2][1]=(1.-gm_bnd_h[l-1])*gm_bub_h[l]/los_h[l];
        mb[1][1]=mb[2][2]=(1.-gm_bnd_h[l])*gm_bnd_h[l-1]/(los_h[l]*los_h[l])+
                                                (1.-gm_bnd_h[l])*gm_bub_hdp[l];
        mb[1][2]=mb[2][1]=2.*(1.-gm_bnd_h[l])*gm_bnd_h[l-
1]*gm_bub_h[l]/(los_h[l]*los_h[l])+
                                                (1.-gm_bnd_h[l])*gm_bub_hd0[l];
        mc[1][1]=mc[2][2]=(1.-gm_bnd_h[l-1])*gm_bnd_h[l]/(los_h[l]*los_h[l])+
                                                (1.-gm_bnd_h[l-1])*gm_bub_hdp[l];
        mc[1][2]=mc[2][1]=2.*(1.-gm_bnd_h[l-
1])*gm_bnd_h[l]*gm_bub_h[l]/(los_h[l]*los_h[l])+
                                                (1.-gm_bnd_h[l-1])*gm_bub_hd0[l];
        md[1][1]=md[2][2]=(1.-gm_bnd_h[l])/(los_h[l]);
        md[1][2]=md[2][1]=(1.-gm_bnd_h[l])*gm_bub_h[l]/los_h[l];
        det=ma[1][1]*ma[2][2]-ma[1][2]*ma[2][1];
        mai[1][1]=ma[2][2]/det;
        mai[1][2]=-1.*ma[1][2]/det;
        mai[2][1]=-1.*ma[2][1]/det;
        mai[2][2]=ma[1][1]/det;
        for (i=1;i<=2;i++)
                for (j=1;j<=2;j++)
                        cma[i][j]=mai[i][j];
        mt=mul_matrices(mai,mb,2);
        for (i=1;i<=2;i++)
                for (j=3;j<=4;j++)
                        cma[i][j]=-1.*mt[i][j-2];
        mt=mul_matrices(mc,mai,2);
        for (i=3;i<=4;i++)
                for (j=1;j<=2;j++)
                        cma[i][j]=mt[i-2][j];
        ma=mul_matrices(mc,mai,2);
        mc=mul_matrices(ma,mb,2);
        mt=sub_matrices(md,mc,2);
        for (i=3;i<=4;i++)
                for (j=3;j<=4;j++)
```

```
                        cma[i][j]=mt[i-2][j-2];
        return cma;
        free_matrix(ma,1,2,1,2);
        free_matrix(mb,1,2,1,2);
        free_matrix(mc,1,2,1,2);
        free_matrix(md,1,2,1,2);
        free_matrix(mai,1,2,1,2);
        free_matrix(mt,1,2,1,2);
        free_matrix(cma,1,4,1,4);
        }

/*
**********************************
        calculate
        cascade matrix of h-pol
                        for layer l

        incorporate small purtubation model
                into the bottom boundary

**********************************
(1)Title        :cas_h_srm(l)
(2)Input parameter
                l(int)   :layer no
(3)Output parameter    : none
(4)Return                               : cascade matrix ( 4 x 4 )
*/

float **cas_h_srm(l)
int l;
        {
        float   **ma,**mb,**mc,**md,**mt,**mai;
        float **cma,det;
        float   gm_btm_0,gm_btm_p;
        int i,j;

        ma=matrix(1,2,1,2);  mb=matrix(1,2,1,2);
        mc=matrix(1,2,1,2);  md=matrix(1,2,1,2);
        mai=matrix(1,2,1,2);  mt=matrix(1,2,1,2);
        cma=matrix(1,4,1,4);

        spm_btm_h(pta[nl],ptb,pa[nl],pab,ang[nl],shs,scl,freq,&gm_btm_0,&gm_btm_p);


        ma[1][1]=ma[2][2]=(1.-gm_bnd_h[l-1])/los_h[l];
        ma[1][2]=ma[2][1]=(1.-gm_bnd_h[l-1])*gm_bub_h[l]/los_h[l];
        mb[1][1]=mb[2][2]=(1.-gm_bnd_h[l])*gm_bnd_h[l-1]/(los_h[l]*los_h[l])+
                                                (1.-gm_bnd_h[l])*gm_bub_hdp[l];
        mb[1][2]=mb[2][1]=2.*(1.-gm_bnd_h[l])*gm_bnd_h[l-
1]*gm_bub_h[l]/(los_h[l]*los_h[l])+
                                                (1.-gm_bnd_h[l])*gm_bub_hd0[l];
        mc[1][1]=mc[2][2]=(1.-gm_bnd_h[l-1])*gm_bnd_h[l]/(los_h[l]*los_h[l])+
                                                (1.-gm_bnd_h[l-1])*gm_bub_hdp[l];
        mc[1][2]=mc[2][1]=2.*(1.-gm_bnd_h[l-
1])*gm_bnd_h[l]*gm_bub_h[l]/(los_h[l]*los_h[l])+
                                                (1.-gm_bnd_h[l-1])*gm_bub_hd0[l]+
                                                (1.-gm_bnd_h[l-
1])*gm_btm_0/(los_h[l]*los_h[l]);
        md[1][1]=md[2][2]=(1.-gm_bnd_h[l])/(los_h[l]);
        md[1][2]=md[2][1]=(1.-gm_bnd_h[l])*gm_bub_h[l]/los_h[l];
        det=ma[1][1]*ma[2][2]-ma[1][2]*ma[2][1];
        mai[1][1]=ma[2][2]/det;
        mai[1][2]=-1.*ma[1][2]/det;
        mai[2][1]=-1.*ma[2][1]/det;
        mai[2][2]=ma[1][1]/det;
        for (i=1;i<=2;i++)
                for (j=1;j<=2;j++)
                        cma[i][j]=mai[i][j];
        mt=mul_matrices(mai,mb,2);
        for (i=1;i<=2;i++)
                for (j=3;j<=4;j++)
                        cma[i][j]=-1.*mt[i][j-2];
        mt=mul_matrices(mc,mai,2);
        for (i=3;i<=4;i++)
                for (j=1;j<=2;j++)
```

```
                        cma[i][j]=mt[i-2][j];
        ma=mul_matrices(mc,mai,2);
        mc=mul_matrices(ma,mb,2);
        mt=sub_matrices(md,mc,2);
        for (i=3;i<=4;i++)
                for (j=3;j<=4;j++)
                        cma[i][j]=mt[i-2][j-2];
        return cma;
        free_matrix(ma,1,2,1,2);
        free_matrix(mb,1,2,1,2);
        free_matrix(mc,1,2,1,2);
        free_matrix(md,1,2,1,2);
        free_matrix(mai,1,2,1,2);
        free_matrix(mt,1,2,1,2);
        free_matrix(cma,1,4,1,4);
        }


/*
*****************************
        calculate
        cascade matrix of v-pol
                        for ice layer

*****************************
(1)Title                :cal_cas_v()
(2)Input parameter      :none
(3)Output parameter :none
(4)Return                       :cascade matrix ( 4 x 4 )
*/

float **cal_cas_v()
        {
        float  **mt1,**mt2;
        int i;
        mt1=matrix(1,4,1,4);
        mt2=matrix(1,4,1,4);
        mt1=cas_v_srm(nl);
        for (i=1;i<nl;i++)
                {
                mt2=mul_matrices(cas_v(nl-i),mt1,4);
                mt1=submatrix(mt2,1,4,1,4,1,1);
                }
        return mt1;
        free_matrix(mt1,1,4,1,4);
        free_matrix(mt2,1,4,1,4);
        }

/*
*****************************
        calculate
        cascade matrix of h-pol
                        for ice layer

*****************************
(1)Title                :cal_cas_h()
(2)Input parameter      :none
(3)Output parameter :none
(4)Return                       :cascade matrix ( 4 x 4 )
*/

float **cal_cas_h()
        {
        float  **mt1,**mt2;
        int i;
        mt1=matrix(1,4,1,4);
        mt2=matrix(1,4,1,4);
        mt1=cas_h_srm(nl);
        for (i=1;i<nl;i++)
                {
                mt2=mul_matrices(cas_h(nl-i),mt1,4);
                mt1=submatrix(mt2,1,4,1,4,1,1);
                }
        return mt1;
        free_matrix(mt1,1,4,1,4);
```

```
        free_matrix(mt2,1,4,1,4);
        }


main()
        {
        int     i,in,ans,nd;
        float  **rc,t1,t2,sig_v,sig_h,**odata,**para;
        odata=matrix(1,18,1,3);
        para=matrix(1,17,1,7);

        /* Input parameters  */

        printf("\nInput Parameters (0:ByHand /1:File)  > ");
        scanf("%d",&ans);
        initParameter();
        if (ans==0)
                {
                inputParameter();
                para[1][1]=freq ; para[1][2]=nl ;
                para[1][3]=pt[0].r; para[1][4]=pt[0].i;
                para[1][5]=ptb.r ; para[1][6]=ptb.i;
                para[1][7]=(float)(bs[0]);
                pta[0]=pt[0];
                for (i=1;i<=nl;i++)
                        {
                        para[i+1][1]=vi[i]*100.0 ; para[i+1][2]=a0[i]*1000.0 ;
                        para[i+1][3]=h0[i]*100.0*2.0 ; para[i+1][4]=d[i]*100.0 ;
                        para[i+1][5]=pt[i].r ; para[i+1][6]=pt[i].i;
                        para[i+1][7]=(float)(bs[i]);
                        }
                para[nl+2][1]=scl*100.0; para[nl+2][2]=shs*1000.0;
                for (i=3;i<=7;i++) para[nl+2][i]=0.0;
                }
        else
                {
                para=readTextSetFile(7,12,&nd);
                freq=para[1][1] ; nl=para[1][2] ;
                pt[0].r=para[1][3]; pt[0].i=para[1][4];
                ptb.r=para[1][5]; ptb.i=para[1][6];
                bs[0]=(int)(para[1][7]);
                pta[0]=pt[0];
                for (i=1;i<=nl;i++)
                        {
                        vi[i]=para[i+1][1]/100.0 ; a0[i]=para[i+1][2]/1000.0 ;
                        h0[i]=para[i+1][3]/(100.0*2.0) ; d[i]=para[i+1][4]/100.0 ;
                        pt[i].r=para[i+1][5] ; pt[i].i=para[i+1][6];
                        bs[i]=(int)(para[i+1][7]);
                        }
                scl=para[nl+2][1]*0.01; shs=para[nl+2][2]*0.001;
                }

        /* Calculation for specific incident angle */

        for (i=1;i<=10;i++)
                {
                in=i*3.0+17.0;
                cal_layer(in);
                rc=matrix(1,4,1,4);
                rc=cal_cas_v();
                t1=rc[4][1]-rc[4][2]*rc[2][1]/rc[2][2];
                t2=rc[1][1]-rc[1][2]*rc[2][1]/rc[2][2];
                sig_v=(1.-gm_bnd_v[0])*t1/t2;
                sig_v=10.0*log10(4.0*Pai*sig_v*cos(Degrad*in));
                rc=cal_cas_h();
                t1=rc[4][1]-rc[4][2]*rc[2][1]/rc[2][2];
                t2=rc[1][1]-rc[1][2]*rc[2][1]/rc[2][2];
                sig_h=(1.-gm_bnd_h[0])*t1/t2;
                sig_h=10.0*log10(4.0*Pai*sig_h*cos(Degrad*in));
                printf("\nAngle = %d / Sigma_v= %f dB / Sigma_h= %f dB",in,sig_v,sig_h);
                odata[i][1]=in; odata[i][2]=sig_v; odata[i][3]=sig_h;
                }

        /* Data Output */
```

```
printf("\n\n Output result to File ? (0:No / 1:Yes)  >");
scanf("%d",&ans);
if (ans==1)
        {
        printf("\n*** Output parameter ***");
        writeTextSetFile(para,7,nl+2);
        printf("\n*** Output data ***");
        writeTextSetFile(odata,3,10);
        }
free_matrix(rc,1,4,1,4);
free_matrix(para,1,nl+2,1,7);
free_matrix(odata,1,18,1,3);
printf("\n\7\7\7 Job Compeleted !!! ");
}
```

```
/ *

        Lake Ice Simulation
                created by Hiro
                                at ASF
                        Created  on     Feb.21,1992
                        Modified on     Mar.23,1992  (add spHmodel and spVmodel)

* /

#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include "dcomplex.h"

#define         Pai     3.1415926
#define Degrad  0.0174533               /* Conversion coefficient degree to radian  */
#define Raddeg  57.29578                /* Conversion coefficient radian to degree  */
#define         Clv     299792458.0     /* Light Speed */


/ *
* * * * * * * * * * * * * * * *
*    sqr      *
* * * * * * * * * * * * * * * *
        Calculate Square value
        1) input parameter
                z  (double)       : value to be squared
        2) return value
                sqr(double) : squared value
* /

double sqr(z)
        double z;

        {
        double d;
        d=z*z;
        return d;
        }


/ *
* * * * * * * * * * * * * * * * * * * * * * *
*   RefractAngle     *
* * * * * * * * * * * * * * * * * * * * * * *
        Calculate Refraction angle
                        Condition:Imaginary part of permittivity is very small.

        1) input parameter
                prmt1 (complex):        relative permittivity for layer 1
                prmt2 (complex):        relative permittivity for layer 2
                prma1 (complex):        relative permiability for layer 1
                prma1 (complex):        relative permiability for layer 2
                theta (float):    incident angle   (degree)

        2) return value
                RefractAngle (float) : refraction angle (degree)
* /

float  RefractAngle(prmt1,prmt2,prma1,prma2,theta)
        dcomplex        prmt1,prmt2,prma1,prma2;
        float           theta;
        {
        dcomplex z1;
        float z;
        z1=Csqrt(Cdiv(Cmul(prmt1,prma1),Cmul(prmt2,prma1)));
        z1=RCmul(sin(Degrad*theta),z1);
        z=asin(Cabs(z1));
        z=Raddeg*z;
        return(z);
        }


/ *
```

```
**********************
*    AttPha       *
**********************
        Calculate Attenuation constant and Phase constant

        1) input parameter
                freq    (float)         : frequency (MHz)
                prmt    (complex)       : relative permittivity for layer

        2) return value
                AttPha (complex)        : Real          --> Attenuation coefficient
                                                            Imaginary --> Phase coefficeint
* /
dcomplex        AttPha(freq,prmt)
        float           freq;
        dcomplex        prmt;
        {
        double  z1,z2;
        dcomplex        z;
        z1=sqrt(sqr(prmt.r)+sqr(prmt.i));
        z2=2*Pai/(Clv/(freq*1.0e6));
        z.r=z2*sqrt(0.5*((-1)*prmt.r+z1));
        z.i=z2*sqrt(0.5*(prmt.r+z1));
        return(z);
        }


/ *
**********************
*   ReflectHpol     *
**********************
        Calculate Reflectivity for H-polarization

        1) input parameter
                prmt1 (complex):        relative permittivity for layer 1
                prmt2 (complex):        relative permittivity for layer 2
                prma1 (complex):        relative permiability for layer 1
                prma1 (complex):        relative permiability for layer 2
                theta (float):          incident angle   (degree)

        2) return value
                calRefHpol (complex) : reflectivity
* /

dcomplex ReflectHpol(prmt1,prmt2,prma1,prma2,theta)
        dcomplex        prmt1,prmt2,prma1,prma2;
        float           theta;
        {
        dcomplex z1,z2,z3,z4,z;
        z1=Cdiv(prmt2,prmt1);
        z2=Cdiv(prma2,prma1);
        z3=Cmul(z1,z2);
        z3.r=z3.r-(sin(Degrad*theta))*(sin(Degrad*theta));
        z3=Csqrt(z3);
        z4=RCmul(cos(Degrad*theta),z2);
        z=Cdiv(Csub(z4,z3),Cadd(z4,z3));
        return z;
        }


/ *
**********************
*   ReflectVpol     *
**********************
        Calculate Reflectivity for V-polarization

        1) input parameter
                prmt1 (complex):        relative permittivity for layer 1
                prmt2 (complex):        relative permittivity for layer 2
                prma1 (complex):        relative permiability for layer 1
                prma1 (complex):        relative permiability for layer 2
                theta (float):          incident angle   (radian)

        2) return value
```

```
                ReflectVpol (complex) : reflectivity
* /

dcomplex ReflectVpol(prmt1,prmt2,prma1,prma2,theta)
        dcomplex        prmt1,prmt2,prma1,prma2;
        float           theta;
        {
        dcomplex z1,z2,z3,z4,z;
        z1=Cdiv(prmt2,prmt1);
        z2=Cdiv(prma2,prma1);
        z3=Cmul(z1,z2);
        z3.r=z3.r-(sin(Degrad*theta))*(sin(Degrad*theta));
        z3=Csqrt(z3);
        z4=RCmul(cos(Degrad*theta),z1);
        z=Cdiv(Csub(z4,z3),Cadd(z4,z3));
        return(z);
        }


/ *
**********************
*    Mixture      *
**********************
        Calculate Dielectric Constant (permittivity)
                        of 2 phase_mixture model

        1) input parameter
                prmth (complex)         :relative permittivity for host material
                prmti (complex)         :relative permittivity for inclusion material
                vi (float)              :inclusion volume factor ( < 0.1 )
                a[]     (float) :depolarization factor
                                                a[0] x-axis
                                                a[1] y-axis
                                                a[2] z-axis
                b[] (float)             :incident angle 3D (degree)
                                                b[0] phai       : azimuth
                                                b[1] theta      : elevation

        2) output parameter
                c[] (complex)   :dielectric constant of 2 phase_mixture model
                                                c[0] x-axis
                                                c[1] y-axis
                                                c[2] z-axis
        3) return value
                Mixture(complex):dielectric constant of incident angle
* /

dcomplex  Mixture(prmth,prmti,vi,a,b,c)
        dcomplex        prmth,prmti,c[3];
        float           vi,a[3],b[2];
        {
        int     i;
        dcomplex z1,z2,z3,d,zz;
        double ex,ey,ez,d1,d2;
        for (i=0;i<3;i++)
                {
                z1=Csub(prmti,prmth);
                z1=RCmul(vi,z1);
                z2=Cdiv(prmti,prmth);
                z2=Csub(z2,Complex(1.0,0.0));
                z2=RCmul(a[i],z2);
                z2=Cadd(Complex(1.0,0.0),z2);
                z3=Cdiv(z1,z2);
                c[i]=Cadd(prmth,z3);
                }
        ex=(sin(Degrad*b[1]))*(cos(Degrad*b[0]));
        ey=(sin(Degrad*b[1]))*(sin(Degrad*b[0]));
        ez=cos(Degrad*b[1]);
        d1=(sqr(ex))*(sqr(c[0].r)-sqr(c[0].i))+
                        (sqr(ey))*(sqr(c[1].r)-sqr(c[1].i))+
                                (sqr(ez))*(sqr(c[2].r)-sqr(c[2].i));
        d2=(sqr(ex))*(2.0*c[0].r*c[0].i)+
                        (sqr(ey))*(2.0*c[1].r*c[1].i)+
                                (sqr(ez))*(2.0*c[2].r*c[2].i);
        d=Complex(d1,d2);
```

```
        d=Csqrt(d);
        return(d);
        }


/*
**********************
*    spHmodel      *
**********************
        Calculate
                backscattering coefficient
                        of small perturbation model in H-polaritzation
        Surface is modeled as Gaussian correlation

(1)Title                :spHmodel
(2)Input parameter
        prmt1 (complex)         :Relative permittivity for layer 1
        prmt2 (complex)         :Relative permittivity for layer 2
        prma1 (complex) :Relative permiability for layer 1
        prma2 (complex) :Relative permiability for layer 2
        theta (int)             :Incident angle (degree)
        s       (float):Standard deviation of surface height
        l       (float):Surface correlation length (meter)
        freq  (float)   :Frequency      (MHz)
(3)Output parameter     :None
(4)Return (float)       :Backscattering coefficient

*/

float spHmodel(prmt1,prmt2,prma1,prma2,theta,s,l,freq)
        dcomplex        prmt1,prmt2,prma1,prma2;
        float           theta,s,l,freq;
        {
        dcomplex c1;
        float     z1,z2,k,w;
        c1=ReflectHpol(prmt1,prmt2,prma1,prma2,theta);
        z1=Cabs(c1)*Cabs(c1);
        k=2*Pai/(Clv/(freq*1.0e6));
        w=0.5*l*l*exp((-1.0)*sqr((k*l*sin(Degrad*theta))));
        z2=8.0*sqr(sqr(k))*(s*s)*sqr(sqr(cos(Degrad*theta)))*z1*w;
        return z2;
        }

/*
**********************
*    spVmodel      *
**********************
        Calculate
                backscattering coefficient
                        of small perturbation model in V-polaritzation
        Surface is modeled as Gaussian correlation

(1)Title                :spVmodel
(2)Input parameter
        prmt1 (complex)         :Relative permittivity for layer 1
        prmt2 (complex)         :Relative permittivity for layer 2
        prma1 (complex) :Relative permiability for layer 1
        prma2 (complex) :Relative permiability for layer 2
        theta (int)             :Incident angle (degree)
        s       (float):Standard deviation of surface height
        l       (float):Surface correlation length (meter)
        freq  (float)   :Frequency      (MHz)
(3)Output parameter     :None
(4)Return (float)       :Backscattering coefficient

*/

float spVmodel(prmt1,prmt2,prma1,prma2,theta,s,l,freq)
        dcomplex        prmt1,prmt2,prma1,prma2;
        float           theta,s,l,freq;
        {
        dcomplex  c1,c2,c3,c4,c5,c6,c7,c8,c9;
        float     z1,z2,k,w,sin2;
        sin2=sin(Degrad*theta)*sin(Degrad*theta);
        c1=Cdiv(prmt2,prmt1);
```

```
c2.r=c1.r-1;
c2.i=c1.i;
c3=RCmul((1+sin2),c1);
c3.r=sin2-c3.r;
c3.i=(-1.0)*c3.i;
c4=Cmul(c2,c3);
c5=RCmul(cos(Degrad*theta),c1);
c6.r=c1.r-sin2;
c6.i=c1.i;
c7=Cadd(c5,Csqrt(c6));
c8=Cmul(c7,c7);
c9=Cdiv(c4,c8);
z1=Cabs(c9)*Cabs(c9);
k=2*Pai/(Clv/(freq*1.0e6));
w=0.5*l*l*exp((-1.0)*sqr((k*l*sin(Degrad*theta))));
z2=8.0*sqr(sqr(k))*sqr(s)*sqr(sqr(cos(Degrad*theta)))*z1*w;
return z2;
}
```

```
/*
*************************************
            readTextFile
*************************************
Read data from Text file
(1) Title                       :readTextFile
(2) Input Parameter     :None
(3) Output Parametar
            n(int)              :Number of data
(4) Return
            data (float):Read data

*/
float *readTextFile(n)
        int             *n;
        {
        char    file[64];
        char    buff[128];
        float   ntemp;
        unsigned int i,nrec;
        FILE    *fp;
        float   *nn;

        printf("\n Text file name    > ");
        scanf("%s",file);
        printf("\n Read data from File : %s\n\n",file);
        if (NULL == (fp = fopen(file,"r")))
                {
                printf("\7\n Cannot Open File : %s\n\n",file);
                exit(1);
                }
        nrec=0;
        while (NULL!=fgets(buff,128,fp))
                {
                nrec++;
                }
        *n=nrec;
        nn=vector(1,nrec);
        rewind(fp);
        for (i=1; i<=nrec ;i++)
                {
                if (NULL == fgets(buff,128,fp))
                        {
                        break;
                        }
                ntemp=atof(buff);
                nn[i]=ntemp;
                }
        fclose(fp);
        return nn;
        free_vector(nn,1,nrec);
        }

/*
*************************************
            readTextSetFile
*************************************
Read data file ( multiple set ) from file (Ascii)
(1) Title                       :readTextSetFile
(2) Input Parameter
            ns      (int)   :Number of data set
            nf      (int)   :Length for one data
(3) Output Parametar
            nd      (int)   :Number of data
(4) Return
            data (float):Read data
*/

float **readTextSetFile(ns,nf,nd)
        int             ns,nf,*nd;
        {
        char    file[64];
        char    buff[256];
        char    obuff[32];
```

```
float    ntemp;
unsigned int i,nrec;
int              j;
FILE     *fp;
float    **nn;

printf("\n Text file name     > ");
scanf("%s",file);
printf("\n Read data from File : %s\n\n",file);
if (NULL == (fp = fopen(file,"r")))
        {
        printf("\7\n Cannot Open File : %s\n\n",file);
        exit(1);
        }
nrec=0;
while (NULL!=fgets(buff,256,fp))
        {
        nrec++;
        }
*nd=nrec;
nn=matrix(1,nrec,1,ns);
rewind(fp);
for (i=1; i<=nrec ;i++)
        {
        if (NULL == fgets(buff,256,fp))
                {
                break;
                }
        for (j=1;j<=ns;j++)
                {
                strpick(buff,(j-1)*nf+j,nf,obuff);
                ntemp=atof(obuff);
                nn[i][j]=ntemp;
                }
        }
fclose(fp);
return nn;
free_matrix(nn,1,nrec,1,ns);
}


/*
*****************************
        writeTextFile
*****************************
write the data(single) to file (Ascii)
(1) Title                       :writeTextFile
(2) Input Parameter
                        data[] (float)   :data
                        n                (int)    :Number of data
(3) Output Parametar:None
(4) Return                      :None
*/

void writeTextFile(data,n)
        float *data;
        int     n;
        {
        char    file[64];
        int             i;
        FILE    *fp;

        printf("\n Destination File     > ");
        scanf("%s",file);
        printf("\n Destination File  : %s\n\n",file);
        if (NULL == (fp = fopen(file,"w")))
                {
                printf("\7\n Cannot Open File : %s\n\n",file);
                exit(1);
                }
        for (i=1;i<=n;i++)
                {
                fprintf(fp,"%12.8f\n",data[i]);
                }
        fclose(fp);
```

```
        }


/*
*****************************
        writeTextSetFile
*****************************
write the data (multiple set) to file (Ascii)
(1) Title                       :writeTextSetFile
(2) Input Parameter
                                data[][](float)  :data
                                m                (int)    :Number of set
                                n                (int)    :Number of data
(3) Output Parametar:None
(4) Return                      :None
*/

void writeTextSetFile(data,m,n)
        float **data;
        int     m,n;
        {
        char    file[64];
        int             i,j;
        FILE    *fp;

        printf("\n Destination File     > ");
        scanf("%s",file);
        printf("\n Destination File  : %s\n\n",file);
        if (NULL == (fp = fopen(file,"w")))
                {
                printf("\7\n Cannot Open File : %s\n\n",file);
                exit(1);
                }
        for (i=1;i<=n;i++)
                {
                for  (j=1;j<=m-1;j++)
                        {
                        fprintf(fp,"%12.6f\t",data[i][j]);
                        }
                fprintf(fp,"%12.8f\n",data[i][m]);
                }
        fclose(fp);
        }


/*
*****************************
        Matrix Calculation
*****************************
Multiply, Subtract, Add matrices
(1) Title                       :mul_matrices, sub_matrices, add_matrices
(2) Input Parameter
                                a,b     (float)         :Matrices a[1..m,1..m], b[1..m][1..m]
                                m       (int)           :Number of row and column
(3) Output Parametar:None
(4) Return                      :Calculated result      [1..m][1..m]
*/

float **mul_matrices(a,b,m)
float   **a,**b;
int             m;
        {
        float  c,**z;
        int  i,j,k;
        z=matrix(1,m,1,m);
        for (i=1;i<=m;i++)
                {
                for  (j=1;j<=m;j++)
                        {
                        c=0.0;
                        for (k=1;k<=m;k++)
                                c=c+a[i][k]*b[k][j];
                        z[i][j]=c;
                        }
```

```
                }
        return z;
        free_matrix(z,1,m,1,m);
        }

float **sub_matrices(a,b,m)
float   **a,**b;
int             m;
        {
        float  **z;
        int  i,j;
        z=matrix(1,m,1,m);
        for (i=1;i<=m;i++)
                for (j=1;j<=m;j++)
                        z[i][j]=a[i][j]-b[i][j];
        return z;
        free_matrix(z,1,m,1,m);
        }

float **add_matrices(a,b,m)
float   **a,**b;
int             m;
        {
        float  **z;
        int  i,j;
        z=matrix(1,m,1,m);
        for (i=1;i<=m;i++)
                for (j=1;j<=m;j++)
                        z[i][j]=a[i][j]+b[i][j];
        return z;
        free_matrix(z,1,m,1,m);
        }
/* polint.c */

#include <math.h>

void  polint(xa,ya,n,x,y,dy)
float  xa[],ya[],x,*y,*dy;
int n;
{
        int  i,m,ns=1;
        float  den,dif,dift,ho,hp,w;
        float  *c,*d,*vector();
        void  nrerror(),free_vector();

        dif=fabs(x-xa[1]);
        c=vector(1,n);
        d=vector(1,n);
        for (i=1;i<=n;i++) {
                if ( (dift=fabs(x-xa[i])) < dif) {
                        ns=i;
                        dif=dift;
                }
                c[i]=ya[i];
                d[i]=ya[i];
        }
        *y=ya[ns--];
        for (m=1;m<n;m++) {
                for (i=1;i<=n-m;i++) {
                        ho=xa[i]-x;
                        hp=xa[i+m]-x;
                        w=c[i+1]-d[i];
                        if ( (den=ho-hp) == 0.0) nrerror("Error in routine POLINT");
                        den=w/den;
                        d[i]=hp*den;
                        c[i]=ho*den;
                }
                *y += (*dy=(2*ns < (n-m) ? c[ns+1] : d[ns--]));
        }
        free_vector(d,1,n);
        free_vector(c,1,n);
}

/* qromb.c */
```

```
#include <math.h>

#define EPS 1.0e-6
#define JMAX 20
#define JMAXP JMAX+1
#define K 5

float qromb(func,a,b)
float a,b;
float (*func)();
{
        float ss,dss,trapzd();
        float s[JMAXP+1],h[JMAXP+1];
        int j;
        void polint(),nrerror();

        h[1]=1.0;
        for (j=1;j<=JMAX;j++) {
                s[j]=trapzd(func,a,b,j);
                if (j >= K) {
                        polint(&h[j-K],&s[j-K],K,0.0,&ss,&dss);
                        if (fabs(dss) < EPS*fabs(ss)) return ss;
                }
                s[j+1]=s[j];
                h[j+1]=0.25*h[j];
        }
        nrerror("Too many steps in routine QROMB");
}


/* trapzd.c */

#undef EPS
#undef JMAX
#undef JMAXP
#undef K
#define FUNC(x) ((*func)(x))

float trapzd(func,a,b,n)
float a,b;
float (*func)(); /* ANSI: float (*func)(float); */
int n;
{
        float x,tnm,sum,del;
        static float s;
        static int it;
        int j;

        if (n == 1) {
                it=1;
                return (s=0.5*(b-a)*(FUNC(a)+FUNC(b)));
        } else {
                tnm=it;
                del=(b-a)/tnm;
                x=a+0.5*del;
                for (sum=0.0,j=1;j<=it;j++,x+=del) sum += FUNC(x);
                it *= 2;
                s=0.5*(s+(b-a)*sum/tnm);
                return s;
        }
}


/* dcomplex.c
This routine is a modified version of complex.c in NRC
 ( modified in order to handle double precision complex )
* /


#include <math.h>

typedef struct DCOMPLEX {double r,i;} dcomplex;
```

```
dcomplex Cadd(a,b)
dcomplex a,b;
{       dcomplex c;
        c.r=a.r+b.r;
        c.i=a.i+b.i;
        return c;
}

dcomplex Csub(a,b)
dcomplex a,b;
{       dcomplex c;
        c.r=a.r-b.r;
        c.i=a.i-b.i;
        return c;
}

dcomplex Cmul(a,b)
dcomplex a,b;
{       dcomplex c;
        c.r=a.r*b.r-a.i*b.i;
        c.i=a.i*b.r+a.r*b.i;
        return c;
}

dcomplex Complex(re,im)
double re,im;
{       dcomplex c;
        c.r=re;
        c.i=im;
        return c;
}

dcomplex Conjg(z)
dcomplex z;
{       dcomplex c;
        c.r=z.r;
        c.i = -z.i;
        return c;
}

dcomplex Cdiv(a,b)
dcomplex a,b;
{       dcomplex c;
        double r,den;
        if (fabs(b.r) >= fabs(b.i)) {
                r=b.i/b.r;
                den=b.r+r*b.i;
                c.r=(a.r+r*a.i)/den;
                c.i=(a.i-r*a.r)/den;
        } else {
                r=b.r/b.i;
                den=b.i+r*b.r;
                c.r=(a.r*r+a.i)/den;
                c.i=(a.i*r-a.r)/den;
        }
        return c;
}

double Cabs(z)
dcomplex z;
{       double x,y,ans,temp;
        x=fabs(z.r);
        y=fabs(z.i);
        if (x == 0.0)
                ans=y;
        else if (y == 0.0)
                ans=x;
        else if (x > y) {
                temp=y/x;
                ans=x*sqrt(1.0+temp*temp);
        } else {
                temp=x/y;
                ans=y*sqrt(1.0+temp*temp);
        }
```

```
        return ans;
}

dcomplex Csqrt(z)
dcomplex z;
{       dcomplex c;
        double  x,y,w,r;
        if ((z.r == 0.0) && (z.i == 0.0)) {
                c.r=0.0;
                c.i=0.0;
                return c;
        } else {
                x=fabs(z.r);
                y=fabs(z.i);
                if (x >= y) {
                        r=y/x;
                        w=sqrt(x)*sqrt(0.5*(1.0+sqrt(1.0+r*r)));
                } else {
                        r=x/y;
                        w=sqrt(y)*sqrt(0.5*(r+sqrt(1.0+r*r)));
                }
                if (z.r >= 0.0) {
                        c.r=w;
                        c.i=z.i/(2.0*w);
                } else {
                        c.i=(z.i >= 0) ? w : -w;
                        c.r=z.i/(2.0*c.i);
                }
                return c;
        }
}

dcomplex RCmul(x,a)
double x;
dcomplex a;
{       dcomplex c;
        c.r=x*a.r;
        c.i=x*a.i;
        return c;
}
```

inches 1 2 3 4 5 6 7 8

cm 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

Kodak Color Control Patches

© Kodak, 2007 TM: Kodak

Blue Cyan Green Yellow Red Magenta White 3/Color Black

Kodak Gray Scale

C Y M

© Kodak, 2007 TM: Kodak

A 1 2 3 4 5 6 **M** 8 9 10 11 12 13 14 15 **B** 17 18 19