Fakultät für Physik und Astronomie

Ruprecht-Karls-Universität Heidelberg

Diplomarbeit
im Studiengang Physik

vorgelegt von
Hannes Konrad
aus Heidelberg

2011

Characterization of the age distribution and the flow field of an Alpine glacier by a combination of simple flow modeling and ground-penetrating radar

This diploma thesis has been carried out by Hannes Konrad at the Institute of Environmental Physics, University of Heidelberg, under the supervision of PD Dr. Olaf Eisen

Characterization of the age distribution and the flow field of an Alpine glacier by a combination of simple flow modeling and ground-penetrating radar:

This thesis deals with exploring the three-dimensional internal age distribution of the small cold glacier saddle Colle Gnifetti. The comparison of the age-depth relations of four cores down to bedrock and the identification of their depth-dependent catchment area upstream of the cores on the respective flow lines are of special interest. Tracking isochronous reflection horizons detected by ground-penetrating radar (GPR) and comparing the core ages via these reflections yield a coherent dating scenario up to 80 years before present. However, this method solely based on ice cores and GPR profiles is limited to shallow depth and thus recent ages due to the lack of clear GPR reflections from internal layering beyond the firn-ice transition. This limitation is overcome by a novel approach including simple 2.5-dimensional flow modeling. The age estimation derived from the flow model is based on trajectory tracking in a two-dimensional velocity field. The GPR profiles are not only used to derive model input, but also to validate the approach: The modeled isochrones are compared to those from GPR up to ages of 44 years providing a general agreement in shape and featuring a potential offset in depth of 0-2 m. Due to the good agreement, the method is extended both laterally and to greater depth: Isochronous surfaces are interpolated from the model output within the drilling area. The ages of these isochrones are compared at the drilling sites and found to cohere up to 120 years. The age coherency between two individual ice cores reaches 200 years. The uncertainty in the age assigned to the isochronous surfaces based on ice core datings increases from 10 years close to the surface to more than 80 years in about 25-60m core depth ($\hat{=}$ 40-60% of ice thickness).

Charakterisierung der Altersverteilung und des Fließfeldes eines alpinen Gletschers mittels einer Kombination aus einfacher Fließmodellierung und Bodenradar:

Diese Diplomarbeit behandelt die dreidimensionale Altersverteilung des kleinskaligen, kalten Gletschersattels Colle Gnifetti. Besonderes Augenmerk liegt auf dem Vergleich der Alters-Tiefen-Beziehungen von vier Bohrkernen, die bis zum Felsbett des Gletschers reichen, sowie auf der Bestimmung des Einzugsbereichs dieser Kerne auf den zugehörigen Fließlinien. Mittels der Identifikation isochroner Reflexionshorizonte durch Bodenradar (ground-penetrating radar, GPR) und dem Vergleich der Bohrkernalter anhand dieser Reflexionen ergibt sich ein konsistentes Datierungsszenario, das die vergangenen 80 Jahre umfasst. Diese Methode, die sich nur auf Daten von Eiskernen und GPR-Profile stützt, ist jedoch auf geringe Tiefen und entsprechend junge Alter beschränkt, da sich unterhalb des Firn-Eis-Übergangs keine eindeutigen internen Reflektoren identifizieren lassen. Ein neuer Ansatz kann dieses Problem durch die Einbindung von 2,5-dimensionaler Fließmodellierung umgehen: Hier erfolgt die Abschätzung des Alters durch das Nachfahren der Trajektorien in einem zweidimensionalen Geschwindigkeitsfeld. Die GPR-Profile werden nicht nur verwendet, um Input für das Modell zu erarbeiten, sondern auch um die Ergebnisse der Methode zu überprüfen: In einer Altersspanne von bis zu 44 Jahren werden die modellierten Isochrone mit denen aus GPR-Profilen verglichen und zeigen eine generelle Übereinstimmung in ihrer Form sowie einen Tiefenversatz von 0-2 m. Aus diesem Grund wird die Methode sowohl horizontal als auch in die Tiefe ausgeweitet. Auf der Grundlage des Modell-Outputs werden die isochronen Oberflächen innerhalb des untersuchten Gebietes interpoliert. An den Bohrstellen werden die Alter dieser Isochrone verglichen und weisen eine Übereinstimmung bis zu einem Alter von 120 Jahren auf; im Vergleich zweier bestimmter Kerne werden 200 Jahre erreicht. Die Unsicherheit der aus den Eiskerndatierungen abgeleiteten Isochronenalter nimmt dabei zu von 10 Jahren nahe der Oberfläche bis zu über 80 Jahren in etwa 25-60 m Kerntiefe zu ($\hat{=}40-60\%$ der Eisdicke).

Contents

1 Introduction

Any finding on the anthropogenic impact on climate needs to be assessed against the background of natural climate variability. Studies of pre-industrial climatic conditions and variations rely on natural archives of paleoclimate such as sea sediments, stalagmites, tree rings or ice masses. In this context, polar ice sheets provide the unique opportunity to study past temperature variations over several glacial–interglacial cycles in combination with the respective greenhouse gas concentrations stored in air bubbles which are entrapped in the ice body [Siegenthaler et al., 2005]. This archive can be accessed by ice cores. The interpretation of signals related to climate requires a chronological and therefore stratigraphical archiving of precipitation. Then, climate can be studied over up to 800,000 years by drilling ice cores on polar ice caps [Lambert et al., 2008; Lüthi et al., 2008].

However, ice cores from polar regions provide information about climate variations only remote from human settlements. The small temperature variations during the Holocene cannot be obtained unambiguously from polar ice cores [Fischer et al., 1998]. Therefore, glaciers, e.g. in the European Alps, are considered to provide supplementary climate information. According to the demand for an archive's stratigraphy, suitable drilling sites are restricted to the high summit region so that the ice bodies are cold and the chronology is not disturbed by melt water and the loss of annual layers. In the European Alps, Col du Dôme (Mont Blanc), Fieschersattel (Bernese Alps), Col del Lys and Colle Gnifetti (both Monte Rosa) meet this requirement [Oeschger et al., 1977].

Colle Gnifetti is unique among this set of cold glaciers: Large amounts of the annual snow precipitation are removed from the surface by strong winds. This leads to low net surface accumulation rates and therefore long-term records clearly exceeding instrumental data sets despite the small vertical range (on the order of 100 m). The ice core which was drilled most recently on Colle Gnifetti in 2005 is assumed to provide records up to several thousands of years [Bohleber, 2008]. However, ice core studies at Colle Gnifetti are challenged by two major problems:

- The records feature strong depositional noise, i.e. short-time variations which are not related to an actual atmospheric signal [Wagenbach, 1992].

- In contrast to ice cores from polar ice caps or sheets, the dating of Alpine ice cores is hindered by the complex flow field due to the small ranges in horizontal direction which are comparable to the vertical scale. Furthermore, the variation of surface accumulation in space and time affects the dating capability. So ice core records cannot be clearly interpreted due to the unreliable time scale [Wagenbach, 1989].

The first topic is subject to an ongoing multi-core approach on Colle Gnifetti at the Institute of Environmental Physics, University of Heidelberg: The detection of a common signal in the records of four Colle Gnifetti ice cores can enable the separation of signal and noise. Since the datings of the ice cores directly affect the time series of the records to be compared, the latter topic becomes important within this approach. Because of the difficulties in dating, it is a priori not clear that the datings from several cores are coherent, i.e. that the ages obtained from the core datings are equal on an isochronous layer (see below) which connects the cores. Consequently, a consistent dating scenario is needed in the area which contains the cores in order to compare the dating and – in the case of age coherency – afterwards to carry out the multi-core approach for interpreting the Colle Gnifetti records.

In the past, different studies concerning the age distribution on Colle Gnifetti were carried out:

- One-dimensional age distributions at certain locations were obtained by dating ice cores [Schäfer, 1995; Armbruster, 2000; Bohleber, 2008]. As mentioned above, the single core dating is not sufficient and the consistency of several cores has not been proved yet.

- Isochronous layers wich can be tracked in radargrams from ground-penetrating radar (GPR) provide two-dimensional age distributions in combination with ice core datings. Although the method on its own cannot provide ages, the one-dimensional ice core datings (and other properties measured in the cores) can be extended in horizontal directions by GPR profiles [Eisen et al., 2003]. This method is restricted in vertical direction because the tracked horizons vanish at the firn-ice-transition.

- A three-dimensional finite-elements-model was developed for Colle Gnifetti providing a three-dimensional age distribution amongst others, e.g. mainly the establishment of a flow law for firn [Lüthi, 2000; Wagner, 1996].

The mentioned studies do not directly provide information about the age coherency of the ice cores. Prior to the work presented in this thesis, the datings of

the cores which are relevant in this thesis were compared via isochronous layers in GPR profiles. The GPR profiles form a closed course and contain these ice cores. The comparison could be carried out up to ages of ∼50-80 years before present and revealed the coherency of the datings in this range [P. Bohleber, pers. comm. – cf. chapter 2]. This gave rise to the hope that the age coherency can be extended to greater depth or higher ages respectively. However, the extension could not be carried out with the available methods because of the mentioned reflector absence below the firn-ice transition. Therefore, a new approach had to be developed.

In this context, the goal of this study is to combine flow modeling with data obtained by GPR-based isochrone mapping in order to ...

1. ... derive a three-dimensional age distribution for the drilling array on Colle Gnifetti in greater depth of the glacier body which can then be used to check the ice core datings for consistency.

2. ... determine possible locations of snow accumulation upstream of the cores in relation to core depth. This is necessary for estimating the influence of the mentioned spatial variability of the surface accumulation on the age-depth distributions at the drilling sites.

With respect to the limited time and computing efficiency and in contrast to the three-dimensional finite elements models mentioned above, a relatively simple two-dimensional flow model will be deployed for this purpose. It will be based on a simple slab model and compensate the missing third dimension by parametrizing transversal ice flow divergence. Furthermore, the model will be applied to flow lines with known surface boundary conditions such as accumulation rate distribution and topography. The three-dimensional age distribution will be obtained by applying the model to several flow lines and then interpolating the isochronous layers between these flow lines. GPR data are available both on several flow lines and on transverse profiles. Three of them have been acquired especially for the work at this thesis in August 2010. The combination of the model with GPR data will be carried out in the following ways:

- Deriving input data (accumulation rate pattern and glacier bed topography) for the model from GPR data.

- Comparing model isochrones to GPR isochrones on the GPR profiles parallel to flow lines in order to estimate the model's quality.

The foci of the evaluation will be on contemplation of the model's characteristics (sensitivity and output variation), on comparison of the model's output to already available age information from ice cores and GPR isochrones and on the check whether the ice core datings are coherent on the modeled isochrones.

2 Study site: Colle Gnifetti

Geographical setting

Colle Gnifetti is located in the summit region of the Monte Rosa massif in the Valais Alps, being partially situated in Italy and Switzerland. It forms a glacier saddle between Zumsteinspitze and Signalkuppe – two summits of Monte Rosa. The saddle is the uppermost part of the accumulation area of Grenzgletscher at altitudes of 4400-4550 m.

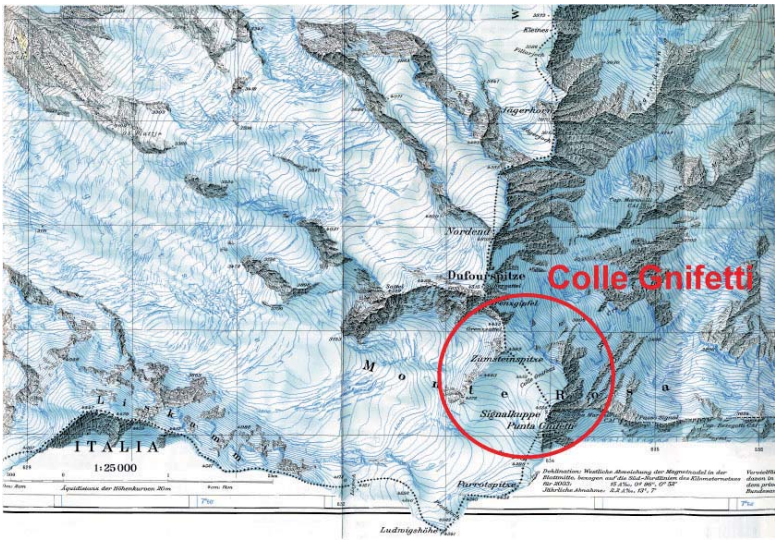


Figure 2.1
Colle Gnifetti within the Monte Rosa massif; from Bohleber [2008].

Overview over the glaciological setting and related studies

One of Colle Gnifetti's main characteristics is the steep ice cliff at the Monte Rosa east face. Wind erosion of the snow, which is blown across the ice cliff and thus permanently removed from the archive, leads to exceptionally low net accumulation rates (0.1-0.5 mwe/a [Wagenbach, 2001]). The accumulation is spatially and temporally irregular and summer biased because the wind erosion preferably affects dry winter snow.

Figure 2.2
Colle Gnifetti: view of the saddle and Signalkuppe from Zumsteinspitze; note the ice cliff on the left; by courtesy of P. Bohleber.

Colle Gnifetti has been intensely studied for the last four decades. While ice core studies will be discussed shortly in an extra section below, a brief overview will be given here:

- **Temperature:** The firn temperature on Colle Gnifetti was determined to be $-15°$ C [Haeberli and Alean, 1985]. Thus, the demand for a cold firn body is clearly met by Colle Gnifetti and the stratigraphy of annual layers can be assumed undisturbed, although there is evidence for a temperature increase in the last decades [Hoelzle et al., 2011]. Furthermore, the glacier bed is frozen to the bedrock due to this low temperature: There is no sliding motion of the base – an important feature for the modeling purpose – and the annual layer thinning with depth is increased.

- **Surface changes:** According to the comparison of photographs from 1893 AD and ∼1994 AD and of geodetic measurements covering the last three decades, the glacier's surface is assumed to be in or at least near steady state [Lüthi, 2000]. Since the model application will be based on the steady state assumption, this is crucial for modeling.

- **Surface velocities:** Velocity measurements by stakes were performed all over the saddle [e.g. Keck, 2001,Lüthi and Funk, 2000]. Some of these have

been used in this thesis (cf. chapter 5). Typical values for surface velocities are 0.5-2 m/a.

- **Flow field modeling:** Colle Gnifetti has been subject to modeling approaches preceding to the three-dimensional approach mentioned in chapter 1 [Haeberli et al., 1988].
- **Ground-penetrating radar:** There have been numerous GPR campaigns on Colle Gnifetti. In recent years, Wagner [1996] and Lüthi [2000] performed measurements in order to obtain boundary conditions for three-dimensional modeling, Eisen et al. [2003] compared ice core data (see chapter 1) and Böhlert [2005] derived a suitable location for the KCI drilling (see below). Thereby, and in campaigns carried out before, the glacier thickness was found to be on the order of 60-100 m.

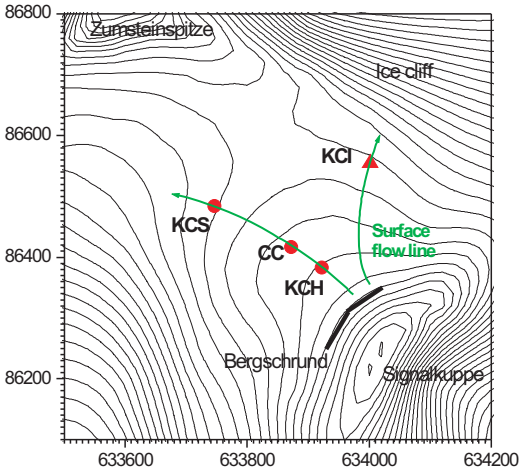


Figure 2.3
Flow lines containing the deep cores; from Bohleber [2008].

core	depth [m]	acc. rate [mwe/a]
KCH	60.3	0.23
CC	64.1	0.22
KCS	100.0	0.51
KCI	61.8	0.14

Table 2.1
Measured ice core depth and accumulation rate (from the tritium peak) of the four deep cores from Schäfer [1995], Armbruster [2000], Bohleber [2008].

Ice cores on Colle Gnifetti

The numerous studies on Colle Gnifetti include drilling a number of ice cores, four of them almost reaching bedrock. These four deep cores were evaluated at the Institute of Environmental Physics and are shown in Figure 2.3. They are situated on two surface flow lines. CC was drilled in 1982, KCS and KCH in 1995. KCI was drilled in 2005 in the saddle region featuring the lowest accumulation rate in order to obtain a long-term record providing climate related time series

over several hundreds or even thousands of years. The two flow lines originate in the same source region near the Bergschrund below Signalkuppe.

Dating Alpine ice cores

There are three main methods for dating the Colle Gnifetti ice cores [Schäfer, 1995; Armbruster, 2000; Bohleber, 2008]:

- **Identification of absolute time markers:** Certain atmospheric disturbances have a characteristic imprint on the snow precipitation at Colle Gnifetti and thus can be used as absolute time markers:
  - higher acidity related to higher sulphate concentrations following prominent volcanic eruptions,
  - Saharian dust blown up to the Alpine summit regions,
  - the tritium peak related to the atmospheric nuclear bomb tests being at its maximum in 1963.

- **Annual layer counting:** Impurity profiles of an ice core can be examined for seasonal cycles. If these are found, they can be used for annual layer counting. This method needs fixed points (time markers – see above) since it only provides time intervals, not explicit ages. Furthermore, this method is restricted in depth by the thinning of annual layers due to vertical strain: At a certain depth, the annual layers cannot be resolved any more.

- **Extrapolation by modeling:** The known time scale from the first two methods can be extrapolated to greater depth by fitting simple flow velocity models (e.g. ice slab – see chapter 3) to the existing data. This approach lacks the accounting for upstream variations in the vertical velocity component. (The results of this thesis concerning the source regions of particles in the ice cores are expected to provide an increase of sophistication in this respect – see chapter 1.)

Each of the mentioned cores on Colle Gnifetti was dated by a combination of at least two of these methods. In core sections, where the absolute time markers can no longer be clearly identified, the error of the age-depth relations increases to unkown extend. Therefore, the datings are subject to revisions. In this context, there has been and will be made effort in deriving further time markers (or at least limitations) at greater depth by ^{14}C-dating [May, 2009].

Dating coherency

In 2000, a GPR profile connecting the KCH and KCS drilling sites was recorded [Eisen et al., 2003]. Additional profiles, which – in combination with the one from 2000 – form a closed course and contain the four cores, were recorded in 2008 (cf. chapter 4). Internal reflection horizons were tracked on this course and the ages of the corresponding depths at the ice core locations were compared [P. Bohleber, pers. comm.]. The results are shown in Figure 2.4. The coherency of the ice core datings is obviously given up to 50 years before present. From there, the data points are only few to a maximum of 80 years due to the absence of clear reflections in greater depth, but they still prove dating coherency. As mentioned in chapter 1, further extension was not possible and is now one of the objectives of the present thesis.

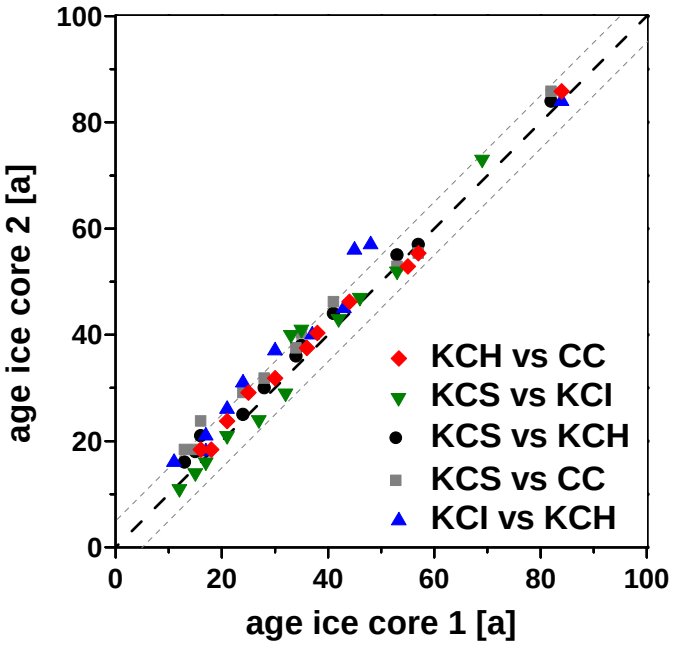


Figure 2.4
Age coherency on tracked internal reflection horizons; the ice core datings are evaluated at the depths of the internal reflection horizons. The corresponding ages of each two ice cores are plotted vs each other. The ages thus cohere if they are situated on the bisecting line. From P. Bohleber, pers. comm.

3 Flow modeling

As outlined in the introduction, the main requirement, which the model used in this thesis has to meet, is simplicity – while still standing on a physical basis of course. In order to keep the computing efficiency and the conceptual effort at a low level, a two-dimensional flow line model has been the object of interest. An adequate basis for this purpose was found in the already existing one presented by Vincent et al. [1997]. The Fortran code was generously provided by Christian Vincent. In the scope of this thesis, several modifications have been applied to this model's concept and performance. This chapter deals first with the general approach to modeling continuum mechanical systems and especially glaciers and then treats simple examples for solving the underlying equations. One of the examples, the so-called ice slab model which is a common approach to glacier flow [Paterson, 1994], serves as a basis for the specific model and is thus modified to meet the requirements of real glaciers. The numerical realization of this modified ice slab model is then explained in the cases of both the original one [Vincent et al., 1997] and the adapted one. Finally, the output characteristics and sensitivity are discussed.

3.1 Introduction to ice dynamics

A look at the mass balance of a glacier provides a first intuitive understanding of why there must be dynamic processes within the firn/ice body: Above a certain altitude (ELA - equilibrium line altitude) there is net accumulation of the yearly snow precipitation. Below the ELA there is net ablation of the glacier's mass. Gravity causes a compensation of this mass accumulation gradient: Mass is transported from the accumulation area to the ablation area by internal deformation which sums up to glacier flow (Figure 3.1). This leads to a velocity field

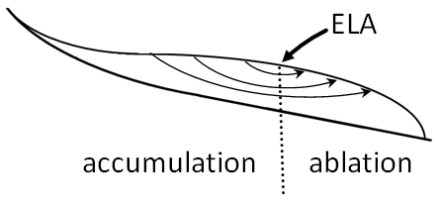


Figure 3.1
Sketch of mass transport in a glacier from the accumulation area to the ablation area with particle trajectories indicated; modified from Paterson [1994].

$\vec{u} = (u,\ v,\ w)^{\mathrm{T}}$ in the glacier. In the case of a glacier in steady state the accumulation gradient is completely compensated and the glacier surface is constant in time.

The age of the ice at a certain point in the glacier equals the local particle's travel time from the surface into the glacier to that point. So the vertical age distribution at any location on the glacier (e.g. a drilling site) is linked to the velocity field. In theory, the age of a particle moving on a trajectory γ from the surface to a point $\vec{r}$ is:

$$t(\vec{r}) = \int_{\gamma} \frac{1}{|\vec{u}(\vec{r}\,'(s))|}\,\mathrm{d}s\ . \tag{3.1}$$

In practice this equation is not evaluable because the velocity field $\vec{u}$ (i.e. the trajectory) is not known. As discussed in chapter 2, one-dimensional (vertical) age distributions at drilling sites from conventional ice core dating feature uncertainties with respect to the detection limit, the neglect of upstream variations in the flow field and the inherent dating uncertainty of Alpine ice cores. Here, flow models provide an alternative approach by simulating the velocity field and calculating the age of the ice by using equation (3.1).

Continuum mechanics as a basis of flow modeling provides balance equations for mass, momentum and internal energy which connect the quantities listed in Table 3.1.

mass:
$$\frac{\partial \rho}{\partial t} + \mathrm{div}\,(\rho\vec{u}) = 0 \tag{3.2}$$

momentum:
$$\mathrm{div}\,\boldsymbol{\tau} + \rho \cdot \vec{g} = \rho\,\frac{\mathrm{d}\vec{u}}{\mathrm{d}t} \approx 0 \tag{3.3}$$

internal energy:
$$\rho\,\frac{\mathrm{d}u_{int}}{\mathrm{d}t} = -\mathrm{div}\,\vec{q}_h + \mathrm{trace}\,(\boldsymbol{\tau}\cdot\dot{\boldsymbol{\epsilon}}) + \rho \cdot p_r \tag{3.4}$$

ρ	density	$\vec{u}$	velocity field
$\boldsymbol{\tau}$	stress tensor	$\vec{g}$	gravitational acceleration
u_{int}	specific internal energy	$\vec{q}_h$	heat flux
$\dot{\boldsymbol{\epsilon}}$	deformation rate tensor	p_r	specific radiation power

Table 3.1
Quantities linked by the balance equations.

Equation (3.4) largely increases the complexity of the system. Following a conventional strategy it will be neglected in this thesis for the sake of simplicity, which

means that energy conservation is not claimed.

The stated balance equations hold for any material. Since there are too many indefinite quantities in the remaining equations (3.2) and (3.3), a constituitive law is needed in order to connect $\boldsymbol{\tau}$ and the deformation rate tensor $\dot{\boldsymbol{\epsilon}}$ (and thus $\vec{u}$ – equation (3.6)). For ice in the relevant stress range (50-200 kPa), it was found by experiment that the relation between $\dot{\boldsymbol{\epsilon}}$ and deviatoric stress tensor[1] $\boldsymbol{\tau}' = \boldsymbol{\tau} - \frac{1}{3}\,\mathrm{trace}\,(\boldsymbol{\tau}) \cdot \mathbb{1}$ can be parametrized as

$$\dot{\epsilon}_{ij} = A \cdot {\tau'_{eff}}^{\,n-1} \cdot \tau'_{ij} \tag{3.5}$$

where τ'_{eff} is the second invariant of $\boldsymbol{\tau}'$ and appears in this context because the constituitive law does not depend on the explicit choice of the coordinate system.

$$\tau'^2_{eff} = \frac{1}{2}\left(\tau'^2_{xx} + \tau'^2_{yy} + \tau'^2_{zz}\right) + \tau'^2_{xy} + \tau'^2_{xz} + \tau'^2_{yz}$$

Equation (3.5) is known as Glen's flow law [Paterson, 1994]. It links the velocity field to $\boldsymbol{\tau}$ because $\dot{\epsilon}_{ij}$ can be written as

$$\dot{\epsilon}_{ij} = \frac{1}{2}\left[\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right] \qquad i,\, j = x,\, y,\, z \qquad \vec{u} = \begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} . \tag{3.6}$$

In the case of ice, a suitable choice of the flow law's exponent determined by measurements is $n = 3$ which yields a non-linear relation. The flow parameter A depends on temperature and other ice properties. Despite its variations it will be considered as constant in the later derivations.

Equations (3.2), (3.3) and (3.5) form a system of coupled partial differential equations that can only be solved analytically or approximately analytically for a limited range of simple geometries. Two of them are presented in the following (sections 3.2 and 3.3).

3.2 Nye's model

Nye's model refers to an ice sheet at its ice divide ($x = 0$) and is a two-dimensional approach (Figure 3.2). The horizontal velocity u is assumed to be independent of depth z. The mass balance of the ice body in steady state then is

$$\frac{\partial\,(u\,H)}{\partial x} = \dot{b} \qquad \text{(cf. equation (3.13))}$$

[1] $\frac{1}{3}\,\mathrm{trace}\,(\boldsymbol{\tau})$ is the hydrostatic pressure. Its subtraction accommodates that hydrostatic pressure proved not to affect the deformation.

where H is the ice thickness and $\dot{b}$ is the accumulation rate. Both are considered to be constant in space (at least in the surrounding area) and time. So it is

$$u(x) = \frac{\dot{b}}{H}\, x \ .$$

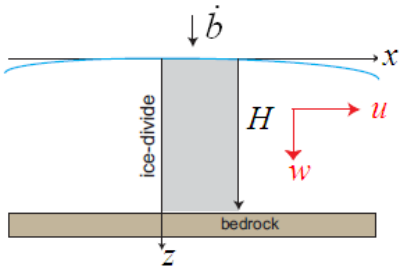


Figure 3.2
An ice sheet/glacier at the ice divide; H is assumed to vary only little in x. Modified from Bohleber [2008].

Ice is a nearly incompressible medium, so the density can also be assumed to be constant by neglecting the firn layer. Equation (3.2) then yields in this two dimensional case:

$$\frac{\partial w}{\partial z} = -\frac{\partial u}{\partial x} \quad \Rightarrow \quad w(z) = \frac{\dot{b}}{H}\,(H - z) \quad \left(w(0) = \dot{b} \text{ and } w(H) = 0 \right) \ . \tag{3.7}$$

The boundary condition $w(H) = 0$ means that the ice sheet is frozen to its bed. $z(t)$ can be calculated using $w = \frac{\mathrm{d}z}{\mathrm{d}t}$ and the age of the ice at depth z can be determined:

$$z(t) = H\left(1 - \mathrm{e}^{-\frac{\dot{b}}{H}\,t}\right) \quad \Rightarrow \quad t(z) = -\frac{H}{\dot{b}} \ln\left(1 - \frac{z}{H}\right) \ . \tag{3.8}$$

Note that the assumption of the vertically constant velocity component u is inconsistent with the assumption that the ice sheet is frozen to the bed. Thus, Nye's model is restricted to upper regions in the ice sheet. It is a purely kinematic approach to ice flow governed by constant vertical strain which is implied in equation (3.7) by $\frac{\partial w}{\partial z} = \text{const}$.

3.3 Ice slab

A simple geometry representing an idealized Alpine glacier is an ice slab with inclined parallel surface and glacier bed (Figure 3.3). The simplifying assumptions and their consequences are:

1. **Infinite extension:** In x- and y- direction there is no limit of the glacier.

$$\Rightarrow \frac{\partial u_i}{\partial x} = \frac{\partial u_i}{\partial y} = 0 \quad \frac{\partial \tau_{ij}}{\partial x} = \frac{\partial \tau_{ij}}{\partial y} = 0$$

Additionally, the symmetry results in zero stress and zero velocity in y-direction, i.e. the problem remains two-dimensional.

2. **Steady state** ($\frac{\partial H}{\partial t} = 0$)**:** This assumption – in combination with the infiniteness and the glacier frozen to bedrock– means that there must not be any accumulation. Since a sink of mass can then only be located at infinite distance, mass accumulation at the surface would lead to a deformation of the surface – in contrast to the steady state assumption.
   The absence of accumulation and ablation results in absence of motion in z-direction ($w = 0$) and in absence of longitudinal stress ($\tau_{xx} = 0$).

3. **Homogeneity:** The density is assumed to be constant all over the ice body. It is synonymous with the demand for incompressibility (cf. 3.2). Equation (3.2) then yields:

$$\operatorname{div} \vec{u} = \frac{\partial u}{\partial x} + \frac{\partial w}{\partial z} = 0 \ . \tag{3.9}$$

   At this stage it is $\frac{\partial w}{\partial z} = 0$ due to the steady state assumption. So the only remaining velocity component is u which does not depend on x. The flow is thus parallel to the surface (laminar flow).

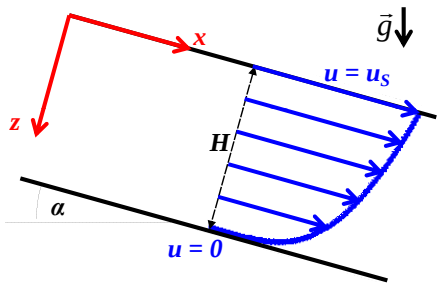


Figure 3.3
Geometric setting of the ice slab;
blue: surface parallel velocity component u;
red: chosen axes parallel and perpendicular to the surface.

The remaining nonzero stress tensor components are τ_{zz} and τ_{xz}. In the flow law the deviatoric stress tensor is considered. Its only component is $\tau'_{xz} = \tau_{xz}$, which is calculated by using the according component of the momentum balance equation (3.3):

$$\frac{\partial \tau_{xz}}{\partial z} = -\rho\, g \sin\alpha \quad \Rightarrow \quad \tau'_{xz} = -\rho\, g \sin\alpha\, z \ ; \quad \tau'_{eff} = \rho\, g \sin\alpha\, z \ .$$

Here, the condition of free surface is imposed ($\tau_{xz} = 0$ at $z = 0$). The flow law (equation (3.5)) then yields

$$\dot{\epsilon}_{xz} = \frac{1}{2}\left(\underbrace{\frac{\partial w}{\partial x}}_{=0} + \frac{\partial u}{\partial z} \right) = -A\,(\rho\, g \sin\alpha\, z)^3 \ . \tag{3.10}$$

The integration can easily be carried out. If the glacier is frozen to its bed, it is $u(H) = 0$. This fixes the constant of integration:

$$u(z) = \frac{A}{2} (\rho\, g \sin\alpha)^3 H^4 - \frac{A}{2} (\rho\, g \sin\alpha)^3 z^4 = u_S \left[1 - \left(\frac{z}{H}\right)^4\right] \tag{3.11}$$

$$\text{where } u_S = \frac{A}{2} (\rho\, g \sin\alpha)^3 H^4 \ .$$

The velocity distribution of equation (3.11) is illustrated in Figure 3.3. For later use the mean velocity component parallel to surface $\overline{u}$ is considered:

$$\overline{u} = \frac{1}{H} \int_0^H u(z)\, \mathrm{d}z = \frac{4}{5}\, u_S \ . \tag{3.12}$$

In this simple situation the complex system of differential equations from section 3.1 could easily be integrated. However the result cannot be applied for dating purposes: The age of the ice is not defined according to equation (3.1) because trajectories run parallel to the surface. This is due to the assumptions which do not meet the circumstances on real glaciers (e.g. Colle Gnifetti). Thus, the ice slab model must be adjusted with respect to

1. **Finite extension:** At least an upper boundary is necessary. While Colle Gnifetti's outflow may be considered to be located at infinity, its upper egde is close the area where the model shall be applied (cf. chapter 2).

2. **Accumulation:** On Colle Gnifetti there is a temporally and spatially variable accumulation (cf. chapter 2).

Furthermore, the situation on Colle Gnifetti demands accounting for

3. **Topography:** In general, glacier bed and surface are neither parallel nor straight.

4. **Variable density:** About the upper half on Colle Gnifetti is firn, which is far from being incompressible. Additionally, firn does not obey the same flow law as ice. The constituitive law for firn is not as simple to parametrize as that for ice (section 3.1) [Lüthi, 2000].

5. **Three-dimensional problem:** The surface flow lines on Colle Gnifetti diverge. So it is clear that two dimensions are not sufficient if there is no alternative treatment of this problem.

The items are listed in order of their ascending complexity. The stepwise modification of the ice slab model according to these items leads to the model used in this thesis.

3.4 Modifications of the ice slab

Uphill limitation and accumulation

If the ice slab is limited in uphill direction, the steady state assumption implies the occurence of accumulation. For the moment, a constant accumulation rate $\dot{b}$ is considered. The velocity components u and w are then expected to depend on x and z. Especially w is expected to be nonzero.

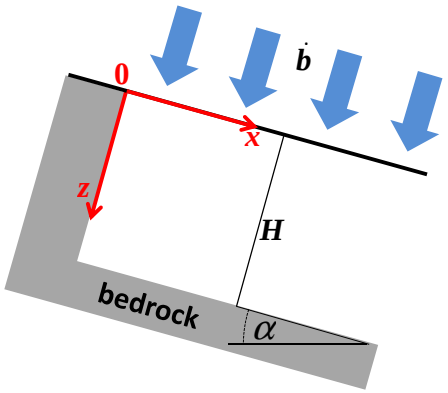


Figure 3.4
The ice slab is limited in uphill direction. A stable surface then needs accumulation.

The glacier's edge is set to $x = 0$ (see Figure 3.4). At a surface point x, the ice flux, which is accumulated uphill from that point, then is

$$q_{acc}(x) = \dot{b}\, x \ .$$

The ice flux through a unit cross section perpendicular to the surface at that x is

$$q(x) = \int_0^H u(x,z)\, \mathrm{d}z = \overline{u}(x)\, H \ .$$

The dependency of u on z is still assumed to be the same as in equation (3.11), i.e. $u(x,z) \propto \left[1 - \left(\frac{z}{H}\right)^4\right]$. Following equation (3.12) it is

$$\overline{u}(x) = \frac{4}{5}\, u_S(x) \quad \Rightarrow \quad q(x) = \frac{4}{5}\, u_S(x)\, H \ .$$

The steady state demands that at every x, the ice flux through the unit cross section equals that accumulated uphill, i.e.:

$$q(x) = q_{acc}(x) \quad \Rightarrow \quad u_S(x) = \frac{5}{4}\,\frac{\dot{b}\, x}{H} \tag{3.13}$$

$$\Rightarrow \quad u(x,z) = \frac{5}{4}\,\frac{\dot{b}\, x}{H}\left[1 - \left(\frac{z}{H}\right)^4\right] \ . \tag{3.14}$$

Via equation (3.9) and the boundary condition $w(x,0) = \dot{b}$, one obtains

$$w(x,z) = \dot{b}\left(1 - \frac{5}{4}\,\frac{z}{H}\left[1 - \frac{1}{5}\left(\frac{z}{H}\right)^4\right]\right) \ . \tag{3.15}$$

In later parts of this thesis, $\dot{b}$ depends on x and the glacier's edge is located somewhere beyond $x = 0$. Then $q(x)$ can be written as

$$q(x) = \int_0^x \dot{b}(\tilde{x})\,\mathrm{d}\tilde{x} + q_0 \ . \tag{3.16}$$

Equations (3.14) and (3.15) still hold – only $\dot{b}\,x$ is substituted by $q(x)$ in (3.14) and $\dot{b}$ by $\dot{b}(x)$ in (3.15).

Variable topography 1 – kinematic correction

Consider an ice slab with non-parallel surface and bed (see Figure 3.5). In formulae (3.14) and (3.15) the ice thickness H then depends on x. Further coupling of this dependency via the original differential equations (3.9) and (3.10) is not taken into account. This is an approximation of zeroth order.

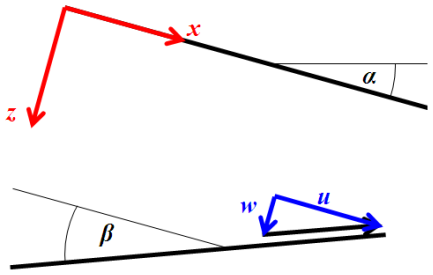


Figure 3.5
Ice slab with non-parallel bed and surface. The kinematic condition is indicated.

Directly at the glacier-bedrock interface, the velocity vector must run parallel to the bed. Otherwise there would be transport of ice mass into or out of the rock. This demand is the kinematic boundary condition. At the bed the velocity components u and w are both zero if the ice is frozen to bedrock. So the kinematic condition is valid for the limit:

$$\lim_{z \to H(x)} \frac{w(x,z)}{u(x,z)} = \tan\beta \ .$$

This condition can be met by adding the so called kinematic correction to w:

$$w_{corr}(x,z) = w(x,z) - \tan\beta\,\frac{z}{H(x)}\,u(x,w) \ . \tag{3.17}$$

Variable topography 2 – piecewise ice slab

The complex surface and bed topography is integrated in the model by the piecewise linearization of the geometry and by the application of the ice slab based equations to these linearized partitions. This demands the introduction of a local coordinate system for piecewise calculation and a global coordinate system (index 0) for input and output (Figure 3.6).

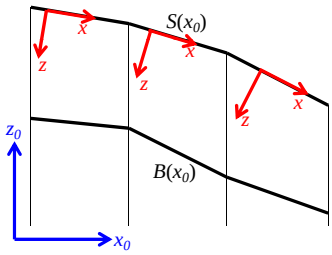


Figure 3.6
Local and global coordinate system in the piecewise application of ice slab;
S is the surface altitude (dependent on x_0).
B is the glacier bed altitude.

This modification demands accounting for the following items:

- The conversion of velocity components u and w from the local system to the global one is a rotation by the angle $\alpha(x_0)$. Since the two systems are oriented contrarily, the signs of the entries of the corresponding rotation matrix are as follows:

$$\begin{pmatrix} u_0 \\ w_0 \end{pmatrix} = \begin{pmatrix} \cos(\alpha(x_0)) & -\sin(\alpha(x_0)) \\ -\sin(\alpha(x_0)) & -\cos(\alpha(x_0)) \end{pmatrix} \cdot \begin{pmatrix} u \\ w \end{pmatrix}; \quad \alpha(x_0) = -\arctan\left(\frac{\partial S}{\partial x_0}\right) .$$

- Since $\tan\beta(x_0) = -\frac{\partial H}{\partial x_0}$, the kinematic correction in formula (3.17) is modified by substituting this expression. ($\beta < 0$ if the bed is steeper than the surface.)

- The allocation of surface points to any point in the glacier on a perpendicular from the surface does not work any more. (In the former system the allocated surface point to every point (x, z) can be found by simply setting $z = 0$.) This is outlined in Figure 3.7. Consequently the following modifications (illustrated in Figure 3.8) in the calculation of u and w are carried out:
  - z is substituted by $Z = S(x_0) - z_0$
  - H is considered vertical ($H(x_0) = S(x_0) - B(x_0)$) instead of perpendicular to the surface ($H_\perp$ in Figure 3.8)
  - $\dot{b}$ and q are evaluated at x_0

  In appendix A.2 the influence of this practice is discussed.

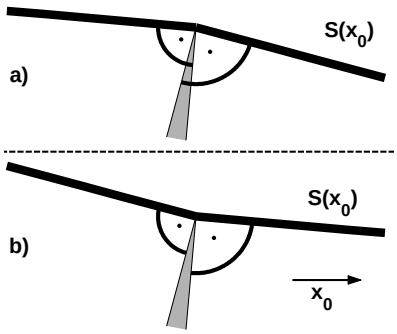


Figure 3.7
a) Concave surface: two surface points can be allocated to every point within the shaded area.
b) Convex surface: there is no surface point to be allocated to the points in the shaded area.

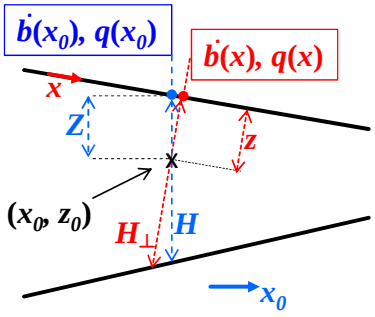


Figure 3.8
Substitution of quantities in the case of an ice slab.

Variable density

Adopting the model to variable density profiles $\rho(x_0, Z)$ is challenging for two reasons:

- The mass balance does not result in div $\vec{u} = 0$. That makes the analytical derivations in section 3.3 invalid.
- The flow law given by equation (3.5) and by the choice of $n = 3$ is only valid in the case of ice. In the upper parts of the glacier a flow law for firn must be considered.

These two problems are circumvented by converting the material to ice equivalent, or water equivalent (we) respectively:[2]

$$Z_{we} = \int_0^Z \frac{\rho(x_0, \tilde{Z})}{\rho_{\text{liquid}}} \, \mathrm{d}\tilde{Z} \qquad H_{we} = \int_0^H \frac{\rho(x_0, \tilde{Z})}{\rho_{\text{liquid}}} \, \mathrm{d}\tilde{Z} \; . \tag{3.18}$$

The velocity is calculated with Z_{we} and H_{we}.
Since the accumulation rate is given in mwe/a, the velocity's unit is also mwe/a. The conversion back to snow/firn/ice is done by scaling the velocity with density:

$$u(x_0, z_0) = u_{\text{firn/ice}} = \frac{\rho_{\text{liquid}}}{\rho(x_0, Z)} \cdot u_{we} \qquad w(x_0, z_0) = w_{\text{firn/ice}} = \frac{\rho_{\text{liquid}}}{\rho(x_0, Z)} \cdot w_{we} \; .$$

[2] The difference between the conversion into ice equivalent and that into water equivalent is a factor that cancels in the formulae.

This is the attempt to account for an additional velocity component due to firn compaction. It is necessarily to be done before the kinematic correction is added because the kinematic condition refers to bed topography, not to ice flow.

Two-dimensional modeling by parametrization of transversal ice flux divergence

The ranges of an Alpine glacier in each direction are approximately of the same order of magnitude (cf. chapter 2). The limitation of the glacier area in transversal direction results in stress and velocity gradients in y-direction and therefore in a nonzero velocity component v. However, the two-dimensional approach is to be kept for the sake of the model's simplicity: A three-dimensional model would necessarily be a finite-elements model and would go far beyond the scope of this thesis. Instead, a new parameter D is introduced, which models transversal ice flux divergence and therefore parametrizes the missing third dimension.

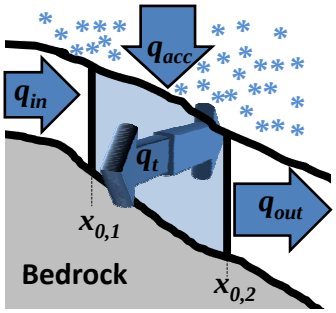


Figure 3.9
Ice flux balance in a box (longitudinal cross section); if there is no basal melting, there is zero flux at the lower boundary (as in the case of Colle Gnifetti, see chapter 2).

If one considers the ice flux balance in a box of a longitudinal (i.e. along a flow line) cross section of the glacier (Figure 3.9), there are four components: q_{in} is the longitudinal inflow at $x_0 = x_{0,1}$, q_{acc} is the accumulated ice flux on the surface, q_{out} is the longitudinal outflow at $x_0 = x_{0,2}$ and q_t is the transversal net ice flux into ($q_t < 0$) or out of ($q_t > 0$) the box. If the flow lines diverge, it is $q_t > 0$. In steady state the mass balance in the box is given by

$$q_{in} + q_{acc} = q_{out} + q_t \ .$$

The parameter D represents that fraction of incoming ice flux that flows to the sides instead of flowing to the right boundary. The balance equation then is:

$$(q_{in} + q_{acc}) \cdot (1 - D) = q_{out} \qquad \left(q_t = (q_{in} + q_{acc}) \cdot D \right) \ .$$

In later computations the box will be defined by the following choices:

$$x_{0,1} = 0 \qquad \Rightarrow \qquad q_{in} = q_0$$
$$x_{0,2} = x_0 \qquad \Rightarrow \qquad q_{acc} = \int_0^{x_0} \dot{b}(\tilde{x}_0)\, \mathrm{d}\tilde{x}_0 \;;\; q_{out} = q(x_0)\ .$$

This leads to

$$q(x_0) = \left(\int_0^{x_0} \dot{b}(\tilde{x}_0)\, \mathrm{d}\tilde{x}_0 + q_0 \right) \cdot (1 - D)\ . \tag{3.19}$$

Note that $q(0) = q_0\,(1 - D) \neq q_0$ if $D \neq 0$ (in contrast to the choice above).
Equation (3.19) allows to apply the ice slab equations and the modifications to flow lines that do not run parallel, which is a typical situation on a glacier saddle.

3.5 Model realization

After having treated the ice slab model and its modifications, the numerical realization is now subject of this section: The model developed and used by Vincent et al. [1997] performs most of the steps of sections 3.3 and 3.4. In the following it will be refered to as COURAN[3]. The version that has been developed within this thesis on the basis of COURAN is called SYNDICATE[4]. These two models are presented with respect to the main tasks, namely input, interpolation routines, trajectory computation and output.

3.5.1 COURAN

Input and interpolation

The necessary input to the model is:

- Surface and glacier bed altitude (S, B) and accumulation rate $\dot{b}$ as a function of horizontal coordinate x_0.
- Density ρ at the two locations $x_{0,D1} = 0$ and $x_{0,D2} = x_{0,max}$ as a function of depth Z, i.e. at an indefinite number of vertical positions Z.
- Divergence parameter $D \in [0, 1)$.

Exemplary input data sets are shown in appendix A.1. S, B and $\dot{b}$ are interpolated linearly between the given data points on a x_0-grid.
The density ρ is first interpolated linearly on a Z-grid at $x_{0,D1}$ and $x_{0,D2}$. Then $\rho(Z)$ is interpolated linearly on the x_0-grid for each Z-value.

[3] According to the name of the passed file.
[4] **SYN**chronizing **D**atings of **I**ce **C**ores by ice sl**A**b based modelling of **T**rajectories and isochron**E**s.

Trajectory computation

Each trajectory starts at a surface point $P^{(0)}$. The velocity at this point is calculated. Then the time step Δt is performed: A particle at $P^{(0)}$ moves to $P^{(1)}$ with the calculated velocity within Δt. This procedure is repeated until (after k time steps) the position $P^{(k)}$ lies outside the defined area. Then a new trajectory is calculated. The model run (including the interpolation routine) is illustrated in Figure 3.13.

Output

The main output file of a model run contains the following data for each point that is located on any trajectory of that run:

- Coordinates (namely x_0 and z_0).
- Velocity in the global reference frame (u_0 and w_0).
- Age t of the ice at that point.

Furthermore, a file is created to which the age-depth distribution at a given location $x_{0,c}$ (subscript c means ice core) is written. The entries of this file are produced whenever a trajectory intersects the model borehole. Exemplary output data sets are shown in appendix A.1.

3.5.2 Syndicate

Couran does not include all the modifications of the ice slab outlined in section 3.4. In order to apply the remaining items Syndicate has been developed on the basis of Couran. Firstly, the code has been checked for consistency and save sources of error have been removed. Secondly, the modifications listed in the following (also illustrated in Figure 3.13) have been applied to the model.

Input

- According to equation (3.16), the additional flux parameter q_0 (to be given in m mwe/a) is introduced. This is a generalization of the q-calculation and makes the code more stable: At $x_0 = 0$, $q_0 = 0$ results in velocity vectors that lead to motion of the particle to $x_0 < 0$ (which is outside the defined area) within the first time step ($u_0 < 0$).
- Additional starting points: If there are points in the considered glacier area at which the age is known (e.g. a dated ice core), these can be handed over

to the model. Trajectories are then calculated not only from surface points, but also from these points.

Density interpolation

- Preceding to the interpolation of density, the depth coordinate Z at $x_{0,D1}$ and $x_{0,D2}$ is converted into the relative depth coordinate Σ:

$$\Sigma = \frac{Z}{H(x_{0,D1/2})} \ .$$

  The density interpolation with respect to x_0 is carried out for given Σ instead of Z as in COURAN. The results of this practice are shown in Figure 3.11.

- $x_{0,D1}$ and $x_{0,D2}$ are free within the range of topography input. At locations $x_0 < x_{0,D1}$ the density ρ is set equal to that at $x_{0,D1}$ for each Σ (and for $x > x_{0,D2}$ respectively).

Velocity computation

While COURAN calculates the velocity components by using depth Z and ice thickness H in meter, SYNDICATE uses these quantities in meter we (Z_{we} and H_{we}) according to the description in section 3.4.

3.5.3 Output processing

There are three objects of interest in the model's output:

- **Trajectories:** They are obtained by simply dividing the output data set into the single trajectories. If one is interested in source regions of a particle, e.g. at a drilling site, one has to rearrange the trajectories with respect to depth at the drilling site.
- **Isochrones:** These are obtained by sorting the output data set with respect to age.
- **Vertical age distribution:** There are two ways of obtaining the vertical age distribution at a location $x_{0,c}$:
  - Via the direct dating output implemented in the code (section 3.5.1).
  - Via the isochrones if $x_{0,c}$ is too close to the edge of the defined area[5] or if several $x_{0,c}$ are considered along one flow line (for k-th isochrone of age t_k: search for the altitude $z_{0,c,k}$ at $x_{0,c}$).

[5] If the trajectory passes not only the borehole but also this edge within one time step, no dating information is written to the according file.

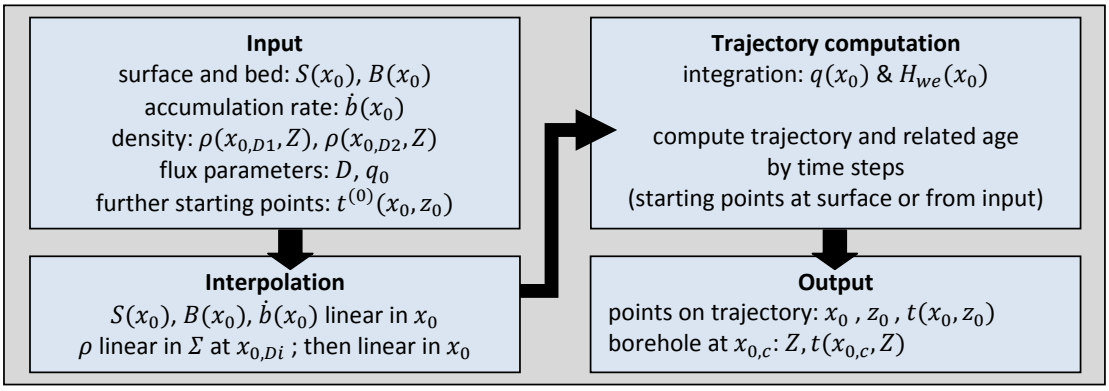


Figure 3.10
SYNDICATE: model run; more detailed: Figure 3.13 (including comparison to COURAN).

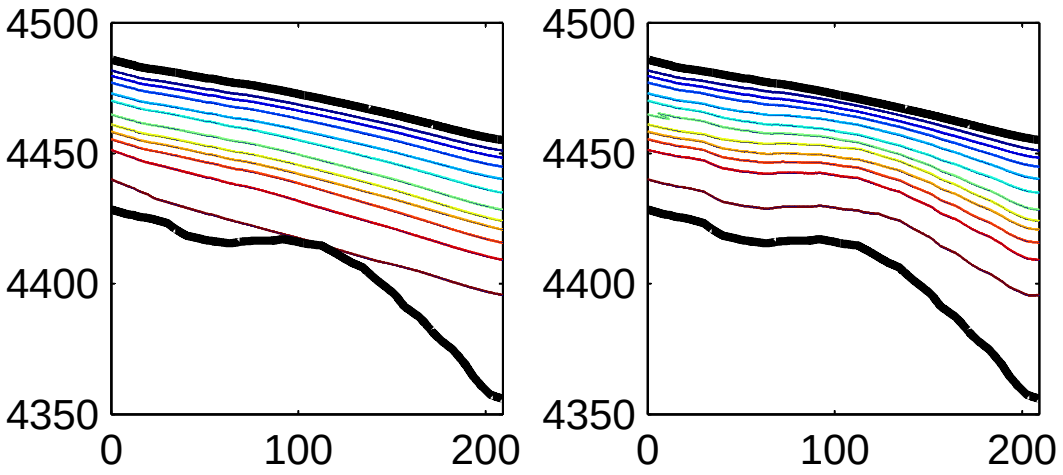


Figure 3.11
Isolines of interpolated density distribution (here: flow line from KCH to KCS); left: COURAN - isolines close to the bed are not parallel to it. Right: SYNDICATE - isolines close to the bed run parallel to it. This guarantees firn-ice-transition in a similar fraction of ice thickness all over the profile as observed in GPR data [Eisen et al., 2003]. COURAN's performance of density interpolation leads to firn-ice transition located too low or even located within the bed rock. In this respect, SYNDICATE's performance of density interpolation is glaciologically more adequate.

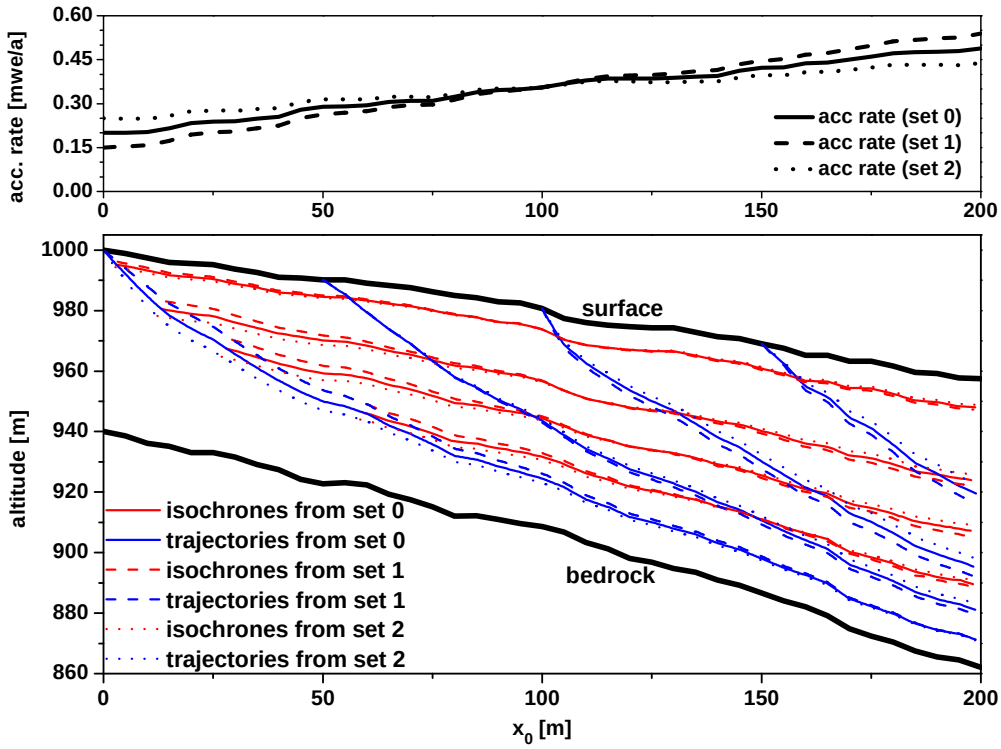


Figure 3.12
Sensitivity to accumulation rate variation; top: 3 different sets of accumulation rate input (designated by indices 0, 1, 2); bottom: corresponding output (line styles equal the input illustration); the remaining input data are listed in appendix A.3 (index 0).
The illustrated trajectories are those starting at $x_0 = 0$, 50, 100, 150 m. The isochrones are those of age $t = 10$, 50, 100, 200 a.

output category	absolute variation	relative variation	most sensitive to
trajectories	0 – 4 m in altitude	0 – 30% of depth	q_0, D, B
isochrones	0.5 – 2 m in altitude	2 – 10% of depth	$\dot{b}$, ρ, B
ice core dating	0 – 150 a in age	5 – 15%	$\dot{b}$, ρ, B

Table 3.2
Variation of output; data from the exemplary model runs in appendix A.3; comment on trajectories: high relative values correspond to low absolute values, i.e. near the starting point / surface.

3.5.4 Sensitivity and propagation of uncertainty

For the later evaluation of the results a statement about uncertainty is essential. There are two types of uncertainty related to the model results.

1. Conceptual uncertainty

Conceptual deficits due to approximations and simplifications have a strong influence on the quality of the results, which is not quantifiable at this stage and is therefore discussed in context with the results in section 6.4.

2. Propagation of input uncertainty

The different responses to variable input have much influence on the interpretation of the model results. This influence is investigated by applying SYNDICATE to an exemplary and fictive flow line. The complete sensitivity study is treated in appendix A.3. The results of the sensitivity study are summarized in Table 3.2.

Exemplarily, the output sensitivity to accumulation rate variations is shown in Figure 3.12: The velocity component w is proportional to $\dot{b}$. Thus, the isochrones' variations are shaped very similar to those of the accumulation rate pattern. The trajectories are influenced by $\dot{b}$ via the q-integration. This is why at large distances the effect cancels in the case of the trajectory starting at $x_0 = 0$..

Since there is a specific uncertainty with respect to the input data derived in chapter 5, a tool is necessary to estimate the uncertainty of the output data. So a bootstrapping routine, which runs the model a defined number of times, and an adequate scheme for random variation of input within the ranges of uncertainty have been developed. These are presented in appendix A.4 with focus on input variation and output analysis.

3.5.5 Summary

SYNDICATE is used for the modeling purpose on Colle Gnifetti according to the objective of this thesis. The desired and examined output consists of

- trajectories,
- isochrones,
- vertical age distributions at (virtual) drilling sites.

The application to flow lines on Colle Gnifetti needs input preparation, namely

- surface and bed topography,
- accumulation rate distribution,
- density profiles,
- flux parameters.

Ice thickness and accumulation rate are obtained by processing of GPR data. This is the subject of the next chapter. The input preparation itself is discussed in chapter 5.

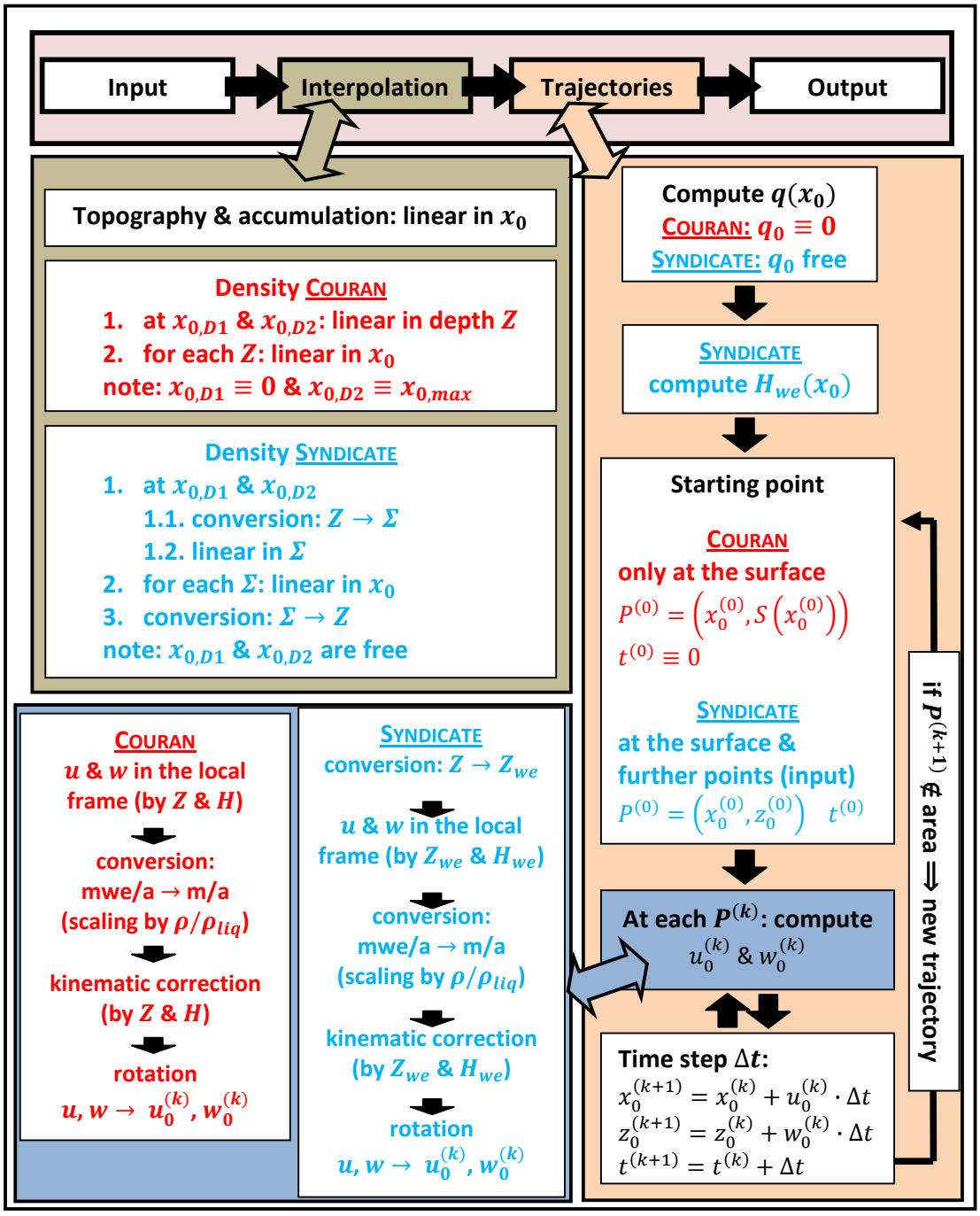


Figure 3.13
Detailed model run (COURAN and SYNDICATE); the time step is $\Delta t = 1a$. The characteristics of COURAN are written in red, those of SYNDICATE in blue.

4 Ground-penetrating radar application

In this chapter, the basics for ground-penetrating radar (GPR) in ice, the acquisition of data and their processing is treated. The general objective at this stage is to derive bedrock topography and dated isochrones from GPR.

4.1 Electromagnetic basics

Electromagnetic waves in materials

For a medium which fulfils the condition of linear and non-dispersive response to applied electromagnetic fields, the Maxwell equations, which are the basis for any description of electromagnetic phenomena, yield wave equations for both the electric and the magnetic field in this medium [Jackson, 1962]. Therefore, energy is transported via propagation of electromagnetic waves, which is the foundation of GPR application.

In vacuum and in isotropic media, the electric field $\vec{E}$, the magnetic flux density $\vec{B}$ and the wave vector $\vec{k}$ pointing in the direction of propagation and energy transport are perpendicular to each other. The wave speed c depends on the medium in which the wave propagates. The following relation holds for the speed of wave propagation, the wavelength λ and the frequency of oscillation ν:

$$c = \lambda \cdot \nu \ .$$

Frequencies of GPR applications are on the order of 100 MHz which is in the radiofrequency spectrum. Electromagnetic (EM) radiation at these frequencies is created by oscillating charge distributions in antennae.

Importance of dielectric properties of a medium to GPR application

In a dielectric medium, an electric field causes the polarization of atomic or molecular dipoles. This weakens the electric field [Jackson, 1962]. The relative permittivity ε is the remaining fraction of the original field strength. In general, $\varepsilon = \varepsilon' - \mathrm{i}\varepsilon''$ is a complex quantity and depends on the frequency of the oscillating field (if this

is oscillating, e.g. in case of a wave). The propagation speed in an isotropic and non-magnetic medium with refraction index $n = n' - \mathrm{i}n'' = \sqrt{\varepsilon}$ is:

$$c = \frac{c_0}{n'} \text{ with} \tag{4.1}$$
$$c_0 = \frac{1}{\sqrt{\varepsilon_0\,\mu_0}} = 299\,792\,458\,\frac{\mathrm{m}}{\mathrm{s}} \text{ (speed of light in vacuum).}$$

Here, ε_0 and μ_0 are the vacuum permittivity and permeability. The imaginary part of n determines the penetration depth α of the wave in the medium:

$$\alpha = \frac{2\omega n''}{c_0} \qquad (\omega = 2\,\pi\,\nu) \qquad \text{[Petrenko and Whitworth, 1999].} \tag{4.2}$$

Note that equation (4.2) is not valid in general but only in the case of low loss media such as ice (see below).

If there is a discontinuity of ε in a medium ($\varepsilon_1 \rightarrow \varepsilon_2$), an incoming EM wave splits into a reflected and a transmitted wave in order to obey continuity conditions. The intensity of the reflected wave (i.e. in principal the transported energy density) is given by

$$I_{refl} = \left(\frac{n_2 - n_1}{n_2 + n_1}\right)^2 I_0 \quad \text{[Demtröder, 2006]}$$

where I_0 is the intensity of the incoming wave. Thus, the difference in n or ε respectively determines the strength of a reflected signal.

In summary, the relative permittivity ε of a medium determines the wave speed and by this the travel times of an emitted radar signal, the absorption of wave intensity and – in combination with a second medium – the strength of reflected signals.

Ice and firn as dielectric

In ice it is $\varepsilon' \gg \varepsilon''$ [Petrenko and Whitworth, 1999]. Therefore, it is classified as low loss medium. Furthermore, it holds $n' \gg n''$, and equation (4.1) yields[6]

$$c = \frac{c_0}{\sqrt{\varepsilon'}}\ . \tag{4.3}$$

[6] $\varepsilon' = n'^2 - n''^2 \qquad \varepsilon'' = 2\,n'\,n''$

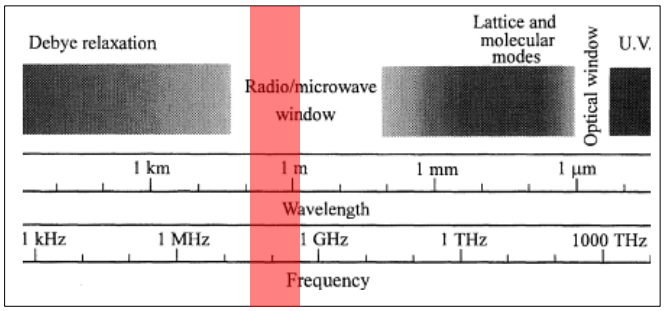


Figure 4.1
Electromagnetic spectrum and relevant processes in ice; the red stripe marks the relevant range for GPR application. Since it is between the resonance frequencies of Debye relaxation (low) and lattice (infrared) and electronic (ultraviolet) oscillations, ε' is almost constant at radar frequencies. Modified from Petrenko and Whitworth [1999].

Although ε' varies within two orders of magnitude at frequencies lower than ~1 MHz, it is nearly constant in the range relevant for GPR application ($\varepsilon' \approx \varepsilon_\infty$ where ε_∞ is the high-frequency permittivity according to the Debye-model [Petrenko and Whitworth, 1999] – Figure 4.1). A typical value for pure polycrystalline ice at temperatures near 0° C is $\varepsilon_\infty = 3.17$ [Fujita et al., 2000], which corresponds to a propagation speed $c_{\text{ice}} \approx 168 \frac{\text{m}}{\mu\text{s}}$.

The hexagonal structure of ice in the Ih-phase causes uniaxial symmetry along the so called c-axis. The prominence of the c-axis therefore results in anisotropy of the optical properties of monocrystalline ice. Thus, especially ε' is a tensor in this case. Dependent on frequency, the relative permittivity parallel to the c-axis $\varepsilon_{||}$ is about 1-20% higher than that transversal to the c-axis $\varepsilon_\perp$ [Fujita et al., 2000]. This anisotropy is usually compensated by the polycristalline structure of ice and firn in glaciers which yields isotropic properties due to the random orientation of the c-axes of the single crystals (for an exception see below).

The relative permittivity of mixtures of two materials can be parametrized by the fractions of each material. Snow and firn can be considered as a mixture of ice and air, for which it was found that the corresponding parametrization is

$$\varepsilon'(\rho) = \left(1 + 0.845 \frac{\text{cm}^3}{\text{g}} \cdot \rho\right)^2 \qquad \text{[Kovacs et al., 1995]} \tag{4.4}$$

where ρ is the firn density and represents the fractions of ice and air. The reported standard error of this formula is $\Delta\varepsilon' = 0.031$.

4.2 GPR as a remote sensing method in glaciology

Reflections of the GPR wave in firn and ice stem from significant changes in dielectric properties. They can be resolved against the noise in the record if the respective changes occur within a distance which is small compared to the wavelength of the incoming signal. These changes can be attributed mainly to variations in density, acidity and crystal orientation fabrics [Fujita et al., 2000]:

- **Density:** According to equation (4.4), changes in ρ, e.g. ice layers in firn resulting from melting and re-freezing events, cause discontinuities in ε'.
- **Acidity:** The acidity affects the solid phase conductivity σ and therefore $\varepsilon'' = \frac{\sigma}{\omega \varepsilon_0}$ [Fujita et al., 2000]. Acidity peaks are related to impurities, e.g. higher sulfate concentration stemming from volcanic eruptions.
- **Crystal orientation fabrics:** Stress induced anisotropic c-axes orientation leads to an ε-discontinuity [Eisen et al., 2007]. The crystal orientation occurs at high stress and thus in greater depth of ice bodies.

Further reflections come from the ice-bedrock interface.

GPR on polar ice sheets

GPR is a well established method for investigating ice sheets. It has been used mainly for ice thickness estimation [e.g. Hempel and Thyssen, 1992]. The data aquisition can be spaceborne, airborne or on sleds. As the firn compaction in ice sheets takes place in a small upper fraction of the ice body, reflectors resulting from density changes are restricted to this region. A further phenomenon of GPR on ice sheets is the radio-echo free zone: In a band of several 100 m thickness above the bed no reflectors are found. Drews et al. [2009] favour the loss of coherent backscatter power due to flow induced layer roughness as the cause of this zone.

GPR on Alpine glaciers

On Alpine study sites, one has to distinguish between GPR application on cold and and on temperate glaciers. The high water content in the latter one strongly influences the recorded signal. Especially concerning Colle Gnifetti as a cold glacier (see chapter 2), GPR has been used since 1980 [Haeberli et al., 1983]. In contrast to ice sheets, crystal orientation has yet not been found to be relevant for reflections [O.Eisen, pers. comm.]. Since the remaining sources of ε-discontinuities – namely density and acidity variations – are commonly considered to stem from

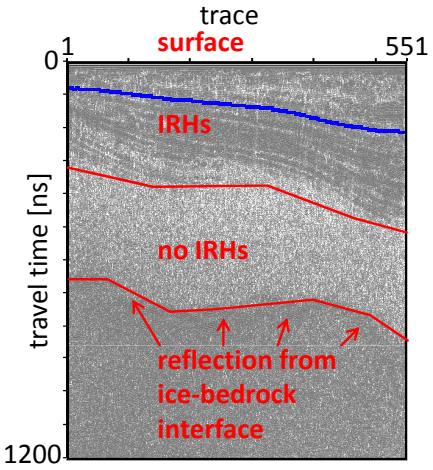


Figure 4.2
Exemplary radargram from Colle Gnifetti (later refered to as T3); the trace indicates the horizontal distance. The grade of shading indicates the reflection's signal strength which is recorded as a function of the signal's travel time. The mentioned features, namely reflection from the bedrock, IRHs (exemplarily, one tracked phase is illustrated in blue) and the region below the firn-ice transition where IRHs vanish, are clearly visible.

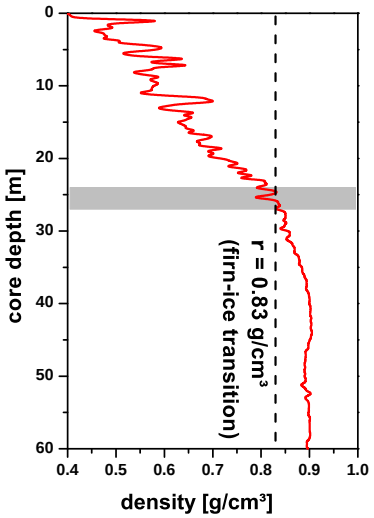


Figure 4.3
Density at KCI; note the peaks above and the absence of peaks below the firn-ice transition (grey region). From the database of IUP Heidelberg.

isochronous events, they can be tracked in order to link ice cores [Eisen et al., 2003] and to estimate patterns of surface accumulation rate [e.g. Böhlert, 2005].

However, IRHs on Colle Gnifetti typically seem to vanish below the firn-ice transition (see Figure 4.2): Density variations can be neglected beyond the transition, since they are on the order of ∼1% in the ice body in contrast to the density variations in firn (Figure 4.3). Eisen et al. [2003] consider that there must also be a lack of strong acidity variations in order to account for the absence of reflectors in this part of the glacier body. A detailed explanation is still topic of ongoing investigations [O. Eisen, pers. comm.]. More in-depth research is necessary including the comparison of ice core based chemical records with their according profiles of dielectric properties [e.g. Jepsen, 2010].

4.3 Data acquisition

Principle of GPR measurements

The GPR setup mainly consists of two antennae: a transmitter (Tx) and a receiver (Rx). The received signal is recorded as a function of time from the emission time. Assuming a constant propagation speed c above an observed plane reflector parallel

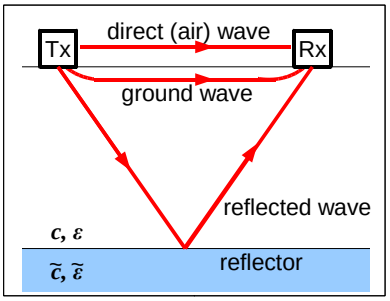


Figure 4.4
Principle of GPR measurement: the transmitter (Tx) emitts a signal. The receiver (Rx) detects the air wave, the ground wave and the reflected wave. Modified from Wagner [1996].

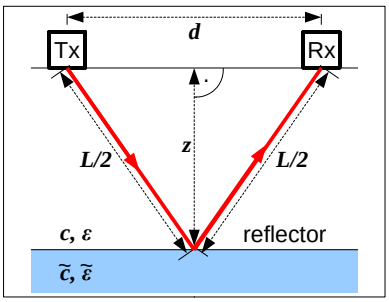


Figure 4.5
Derivation of reflector depth; travel distance of the reflected signal: $L = c\,t_{\mathrm{TWT}}$. TWT means two way travel time. Modified from Wagner [1996].

to the surface, the depth z of this reflector is[7]

$$z = \frac{1}{2}\sqrt{(c\,t_{\mathrm{TWT}})^2 - d^2} \quad \text{(cf. Figure 4.5)}.$$

If the reflector depth z is much greater than the distance d between the antennae (orders: $z \sim 10$ m vs. $d \sim 10$ cm), d can be neglected and it is

$$z = \frac{1}{2}\,c\,t_{\mathrm{TWT}}\ .$$

A vertically variable propagation speed distribution is taken into account by integration:

$$t_{\mathrm{TWT}} = 2\int_0^z \frac{1}{c(\tilde{z})}\,\mathrm{d}\tilde{z}\ . \tag{4.5}$$

Note that this is an implicit relation for $z(t_{\mathrm{TWT}})$.

The common set-ups for GPR measurements are

[7] Note that z does not necessarily refer to a coordinate system parallel and perpendicular to the surface as in chapter 3.

- **Common offset (CO):** The distance between transmitter and receiver is fixed while they are moved along a certain profile. Thus, the point of reflection moves along the reflector.
- **Common midpoint (CMP):** The point of reflection is fixed by symmetrically moving transmitter and receiver in opposite directions. This set-up is used e.g. to determine the vertical propagation speed distribution.
- **Borehole tomography:** One antenna is moved vertically in a borehole, the other one is placed at the surface in order to obtain information about mainly horizontal wave propagation.
- **Vertical radar profiling:** Both antennae are moved vertically with fixed distance (as in CO set-up).

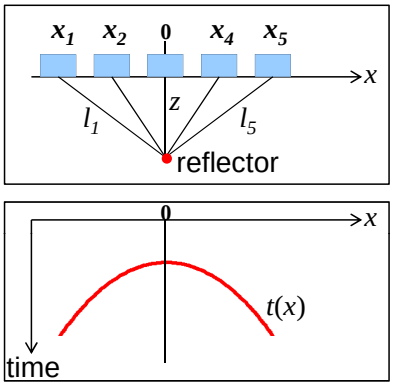


Figure 4.6
Top: distance of the antennae to the reflector in measurement; bottom: resulting hyperbola in the radargram; modified from Böhlert [2005].

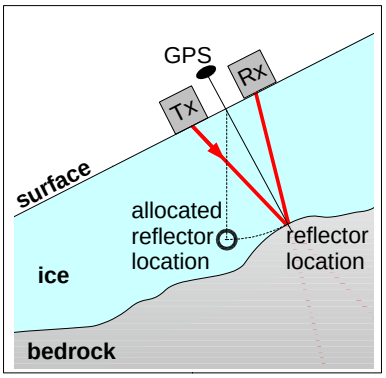


Figure 4.7
Error related to false allocation of reflector location; modified from Böhlert [2005].

On Colle Gnifetti, measurements were carried out in all four set-ups. The GPR data treated in this thesis stem from CO measurements. They feature two difficulties which are linked to the TWT-depth relation:

- Reflected signals come from any reflector, not only from those directly below the antennae. However, the received signals are allocated to the trace and therefore to the horizontal position of the antennae. In the cases of punctual diffractors and edges (in the bedrock), hyperbolae appear in the radargram (Figure 4.6): The antennae distance to the reflector at location x is

$$l(x) = \sqrt{z^2 + x^2} \quad \Rightarrow \quad t_{\mathrm{TWT}}(x) = \frac{2}{c}\sqrt{z^2 + x^2}\ .$$

The aperture angles of these hyperbolae depend on the propagation speed. This problem can be solved by migration which additionally allocates the recorded signal to the corresponding position of the point.

- Apart from the appearance of reflection signals that do not result from reflectors underneath the antennae, the nearest reflections are not necessarily those vertically below the antenna. It comes from that point of the reflecting layer that is located nearest to the antennae. This is shown in Figure 4.7. Since the emission of the signal is a three-dimensional problem, the reflectors might even not be located along the profile which leads to an underestimation of reflector depth. If this is the case, only a three-dimensional migration can correct this effect. This problem is not restricted to CO measurements.

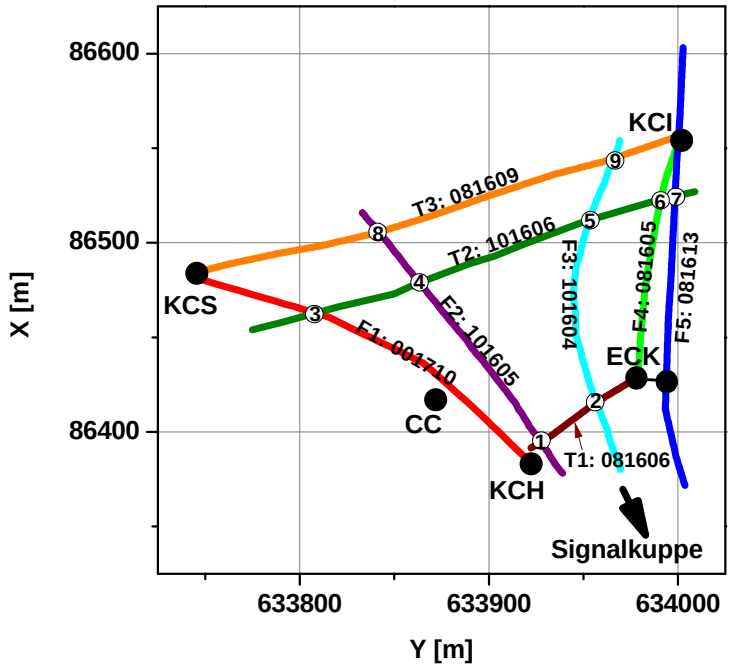


Figure 4.8
GPR profiles evaluated in this thesis; the numerical order is due to the position and not to the time of acquisition. The additional numerical designations are those used in processing. The intersections are designated either by the name of the situated ice core or by numbers. ECK is the designation for the intersection of T1 and F4. The GPS coordinates are converted from WGS84 to Swiss grid.

4.3.1 Evaluated profiles on Colle Gnifetti

Within this thesis, eight CO profiles are considered (Figure 4.8). They are either parallel to flow lines (designation F) or approximately transversal (T) to them. Profile F1 was recorded in 2000 and is one of the subjects in the study carried out by Eisen et al. [2003]. F4, F5, T1 and T3 were measured in 2008 in order to obtain a closed circuit connecting the drilling sites of the deep ice cores. When the work in this thesis made progress the remaining three profiles were recorded in August

2010 in order to improve the spatial resolution at the study site: two further flow line parallel profiles (F2 and F3) and one transversal (T2). Processed GPS data sets for each profile are available [O. Eisen, pers. comm].

Details of data acquisition

The measurements were performed using a RAMAC GPR system from Malå Geoscience, Sweden with shielded 250 MHz antennae. These are placed on a sled in a distance of 36 cm. The wavelet consists of about five half cycles, yielding a total signal length of about 15 ns. This corresponds to a bandwidth of ∼70 MHz, implying that a theoretical separation of reflectors requires that they are separated more than 7.5 ns or approximately 1.3 m.

Each trace (i.e. each record at a certain horizontal distance) is recorded by stacking of eight samples (32 in the case of F1). The shot is distance-triggered by an odometer. In the case of F1, F2, F3, F4, T1 and T3 the trace distance is 0.5 m. On F5, it is 0.2 m. On T2 the triggering by odometer did not work and the shots were triggered manually. This led to an increase in mean trace distance to ∼1.6 m (153 traces on a distance of ∼247 m), but the exact distance between two adjacent traces cannot be reconstructed any more.

4.4 Data processing

In order to obtain isochrones from the GPR data, IRHs have to be identified and digitalized and the corresponding TWTs have to be converted into depth coordinates. This section deals with the necessary steps and procedures on the way to dated isochrones from GPR.

Raw data

The raw data cannot be interpreted reliably and are thus subject to processing routines. Thereby, the allocation of IRHs to their actual situation in the glacier is improved. The processing has been carried out by Paradigm Geophysical FOCUS version 5.4. The used modules are presented in the following. They represent a standard procedure of processing data from seismic and GPR measurements in CO set-up [Yilmaz, 2001; Navarro and Eisen, 2009]. The effect of the processing is illustrated in Figure 4.9.

Static correction: For each trace the recorded signal is shifted in order to have the first break of the air wave at $t_{\mathrm{TWT}} = 0$. The parts before that time are removed.

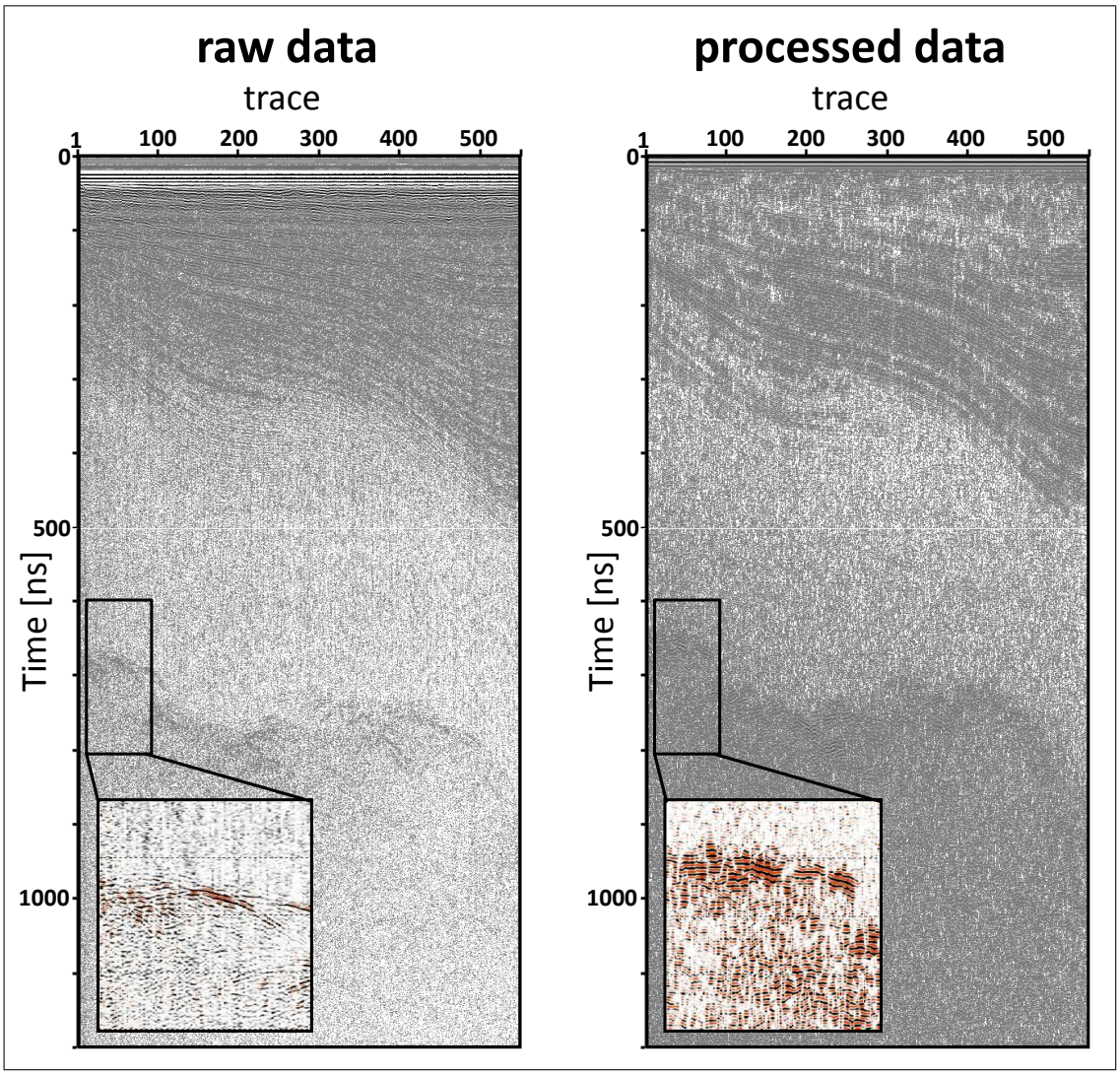


Figure 4.9
Raw and processed radargram (profiles T3); the grade of shading indicates the signal strength. The IRHs and the bedrock reflection ($t_{\mathrm{TWT}} > 650$ ns) are clearly visible. The zoomed section illustrates the migration.

Filter: The signal-to-noise ratio is increased by applying a trapezoidal bandpass filter to the radargram. Consequently, the noise is suppressed and the desired signals at frequencies near 250 MHz become more prominent.

Gain: The spherical divergence of the energy density of the propagating wave at distance d from the transmitter is proportional to d^{-2}. Thus, the intensity of reflections from deep layers is lower than that from layers near the surface. The amplitudes A are corrected by a t_{TWT}-dependent factor in order to correct the divergence:

$$A_{corr}(t_{\mathrm{TWT}}) \propto A(t_{\mathrm{TWT}}) t_{\mathrm{TWT}}^{n} \ .$$

Theoretically, a constant propagation speed would result in $n = 1$ (see above: energy density $\propto$ squared amplitude), but $n = 1.2$ proves to optimize the radargram quality.
Additionally, the illustration of the signals is improved by Automatic Gain Control on some profiles: Each trace is divided into several time windows. Then the amplitude in each window is normalized by the mean amplitude in this window. Consequently, the amplitudes of reflections at different time ranges are illustrated with a similar order of magnitude so that strong reflection do not completely dominate the radargram. This is most important in searching for possible reflections from the ice-bedrock interface (section 4.5.1).

Migration: As already discussed in section 4.3, the reflections are allocated to positions vertically below the antennae although they might come from other directions. The two-dimensional migration allocates the reflections to their true positions assuming that these are situated in the vertical plane containing the profile. Reflections from the sides can only be corrected by three-dimensional migration. Neither is the presented data volume big enough nor are the trace locations known precisely enough for three-dimensional migration. Thus, only two-dimensional migration is carried out. For migration a wave speed distribution has to be provided to the processing software. The wave speed distribution derived from the KCI density distribution via equation (4.4) leads to disappearance of the diffraction hyperbolae in all processed profiles and proves thus suitable.
On profile T2 the migration is rejected because of the large trace distance: The radargram is over-migrated. In addition, the exact trace distance is not known but necessary for migration.

IRH and bedrock reflection picking

The tracking of IRHs is carried out manually. The radargram is displayed with an appropriate zoom factor and one prominent phase of the reflection is picked by

eye. The according trace numbers and TWTs are stored. In the processing, the TWTs are interpolated linearly to each trace between the stored traces.

The traces at the intersections of two profiles or those of the nearest points on two profiles are determined by using the GPS coordinates. Then eleven phases on profile F1, that can be tracked especially well, are chosen. On the remaining profiles the tracking is started on suitable phases at the intersections in order to pick the same reflection layer as on F1. On each profile the picking starts at that intersection trace of the deepest reflections in order to avoid ambiguities from phase divergence. Where the phase cannot be clearly identified, the surrounding phases are used as upper and lower boundaries.

Due to the strong variations of the bedrock topography and roughness, the reflection from the ice-bedrock interface does not display a continuous phase. For its detection, the amplitude's gain is reduced to such degree that signals above the bedrock reflection completely vanish. Then the uppermost visible reflections are picked.

Conversion to depth

Equation (4.5) is used to determine both ice thickness and depth of the IRHs from the corresponding TWTs where the integration is performed as discrete sum. The vertical wave speed distribution is obtained via the density-dependent ε'-parametrization (equation (4.4)). For this reason, a density interpolation at each trace of each GPR profile is necessary. The interpolation applied in this context is the same as used for SYNDICATE (cf. section 3.4). Since the ice thickness is not known at this stage, the density distribution is considered as a function of relative depth $\Sigma \in [0, 1]$. The determination of the density distributions at the two fixed points $x_{0,D1}$ and $x_{0,D2}$ is described in section 5.2.

The ice thickness at each trace can be calculated from the according TWT t^H_{TWT} by substituting $\tilde{z}$ by $\Sigma{\cdot}H$ in equation (4.5):

$$H = \frac{t^H_{\mathrm{TWT}}}{2} \left[\int_0^1 \frac{1}{c(\rho(\Sigma))} \, \mathrm{d}\Sigma \right]^{-1} . \tag{4.6}$$

With known ice thickness, the density ρ can be considered on the absolute depth axis and equation (4.5) is used for converting the TWT of each IRH ($t^{\mathrm{IRH}}_{\mathrm{TWT}}$) to depth ($z^{\mathrm{IRH}}$) by integrating $c(\rho(\tilde{z}))^{-1}$ until $t_{\mathrm{TWT}} = t^{\mathrm{IRH}}_{\mathrm{TWT}}$. Then it is $z^{\mathrm{IRH}} = z$. Furthermore, ice thickness and IRH depth in water equivalent are calculated according to equation (3.18).

IRH dating

The age of each picked IRH can be obtained by considering the ages at the specific depth of the IRH at the drilling sites.

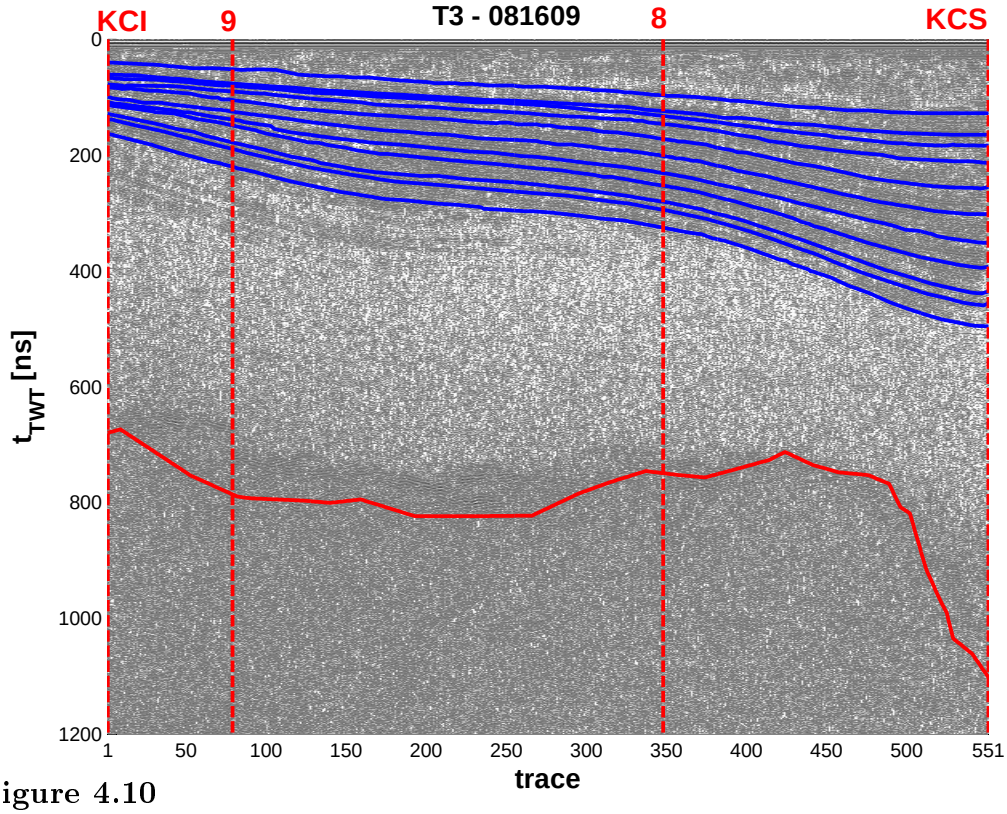


Figure 4.10
Radargram of T3; blue lines: IRHs; red solid line: bedrock reflection; red dashed lines: horizontal positions of intersections with other profiles. The intersection numbers are the same as in Figure 4.8.

4.5 Results of GPR data acquisition and processing

In this section, the results of the described procedure and their errors are presented and discussed. The focus will be set to systematic uncertainties of the method and in the end to the dating of the picked horizons.

4.5.1 Tracking of IRH and bedrock reflections

The radargrams of the eight profiles including the IRHs and the bedrock reflection are presented in appendix B.2. Exemplarily, the one of T3 is shown in Figure 4.10.

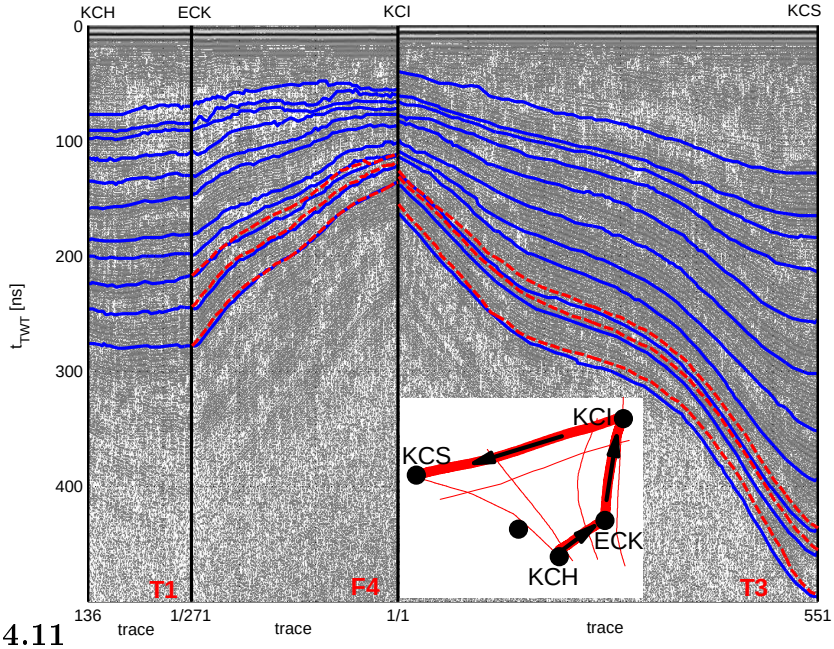


Figure 4.11
Evaluation of picking accuracy and influence of migration: migrated radargrams and picked layers (blue) of T1, F4 and T3; the picking is carried out towards KCI from both directions, starting at KCH/KCS. The three undermost layers differ the most at KCI (∼30 ns). The red dashed horizons are those picked in the unmigrated radargrams. The discrepancy of these horizons is ∼15 ns (≈ 2.5 m). The arrows in the overview follow the illustrated profiles (not the IRH tracking in the radargrams – see above). The horizontal distances along these three profiles amount to ∼480 m.

IRH picking

The vertical resolution of the GPR measurement with an ideal pulse of one cycle length is $\Delta z \approx \lambda$, where λ is the wavelength: The phase from a reflection is only spread over a range of $\lambda/2$, but the extension of the wavelet has to be taken into account as well [Navarro and Eisen, 2009]. According to this depth uncertainty, the error of the picked TWTs due to the spatial extension of the EM wave is $\Delta t^{(1)}_{\mathrm{TWT}} = 4$ ns. A further uncertainty component comes from picking the wrong phase when the tracked phase cannot be identified unambiguously. It thus stems from subjective picking and is not related to conceptual uncertainty as resolution. Typically, the related error amounts to $\Delta t^{(2)}_{\mathrm{TWT}} = 4$ ns. The resulting total uncertainty of the picked TWTs is

$$\Delta t_{\mathrm{TWT}} = \sqrt{\left(\Delta t^{(1)}_{\mathrm{TWT}}\right)^2 + \left(\Delta t^{(2)}_{\mathrm{TWT}}\right)^2} \approx 5 \text{ ns}. \tag{4.7}$$

The TWTs of the eleven horizons at the intersections are given in Table B.1 (appendix B.1). Within the mentioned uncertainty (equation (4.7)), most IRHs fit at the intersections: 10 ns ($\approx$ 1.7 m) of discrepancy are exceeded only in few cases. Problems mainly occur on T2, where the trace distance is too high to track the phases[8], and near KCI, where the lowermost tracked phases on F3, F4 and F5 converge to a higher degree than on T3 (Figure 4.11). The contribution of the migration to that discrepancy is tested by picking according layers in the unmigrated radargrams on T3 and F4. The migration is expected to influence the IRH coherency at the intersection because it changes the TWT (and traces) of the phases and therefore the IRH depth. These changes are not expected to be the same on transversal and longitudinal profiles. The layers picked in the unmigrated radargrams are also shown in Figure 4.11. The migration results in a further depth dependent uncertainty on the order of $\sim$10 ns which is now systematical. This error is not considered further due to the adequate coherency on the upper IRHs. The three lowermost IRHs are rejected because their consistency on the eight profiles is not given.

The mismatch of the TWTs at the intersection has been the reason for numerous revisions and new tracking attempts. The coherency as presented in Table B.1 in appendix B.1 and as discussed above is thus assumed to be the best possible result within this processing routine.

Bedrock reflection picking

Especially the bedrock reflections are influenced by the effects described in section 4.3. Therefore, the picked signal peaks might also come from locations transversal to the profile direction. This is most probably given on the transversal profiles due to the radiation pattern of the transmitter. Consequently – if necessary – prominent deeper reflections on these profiles are picked in order to increase the coherency of all profiles. The TWT error of the picked bedrock reflection is thus estimated to be $\Delta t_{\mathrm{TWT}} = 20$ ns. The problem is illustrated in Figure 4.12 at the KCS drilling site. Note that at KCS the discrepancy is extraordinary high and can thus not be reflected by the assumed TWT error.

Error estimation of IRH depth and ice thickness

The errors of both ice thickness and IRH depth result from two error sources:

- Error of the TWT (see above): $\Delta H_{\mathrm{TWT}}, \Delta z^{\mathrm{IRH}}_{\mathrm{TWT}}$,

[8] For this reason only the seven upper horizons were considered on T2.

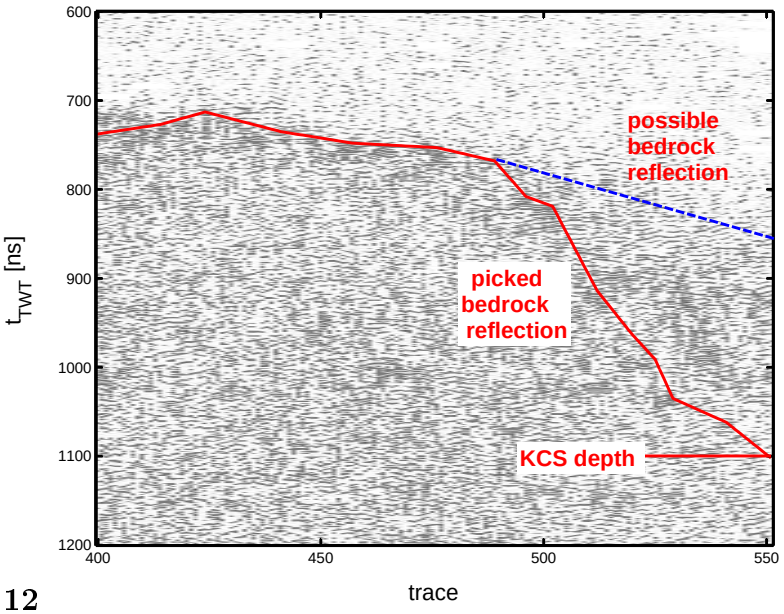


Figure 4.12
KCS on T3; following the method described on page 46, the blue line would have been picked as bedrock reflection, ending at ∼850 ns at KCS, but the depth of KCS (100 m) corresponds to $t_{\mathrm{TWT}} \approx 1100$ ns. Thus, the bedrock reflection is assumed to be that of the red line.

- Error of the wave speed distribution (from the density error - see section 5.2): $\Delta H_c, \Delta z_c^{\mathrm{IRH}}$.

Therefore, the combined errors for H and z^{IRH} are:

$$\Delta H = \sqrt{(\Delta H_{\mathrm{TWT}})^2 + (\Delta H_c)^2} \qquad \Delta z^{\mathrm{IRH}} = \sqrt{(\Delta z_{\mathrm{TWT}}^{\mathrm{IRH}})^2 + (\Delta z_c^{\mathrm{IRH}})^2}\ .$$

A detailed error consideration can be found in appendix B.3. The error of the IRH depth varies in the range of 0.5 m and 0.8 m, the error of the ice thickness is ∼2 m.

4.5.2 Challenges in dating IRHs

Table 4.1 contains the results of the allocation of ages to the IRHs. An improved insight to the results might be gained from Figure 4.13 in which the structure of the results is illustrated on the closed course containing the ice cores. Note that the ice core CC has not been considered at this stage because it is a few meters remote from the GPR profiles. For the relevant IRHs the dating uncertainty is governed by that of the ice core datings ($\Delta t_{\mathrm{dat}}^{\mathrm{dat}} \approx 3$ a [Armbruster, 2000; Bohleber, 2008]).

A further dating uncertainty $\Delta t_z^{\mathrm{dat}}$ results from the depth error as mentioned in section 4.5.1. Then the final error of the IRH dating t^{dat} is:

$$\Delta t^{\mathrm{dat}} = \sqrt{(\Delta t_{\mathrm{dat}}^{\mathrm{dat}})^2 + (\Delta t_z^{\mathrm{dat}})^2} \ .$$

The following is found:

- Within the uncertainty interval, the IRH ages from KCI and KCS on T3 fit very well.
- The age coherency with KCH is not given as good as between KCI and KCS (systematically higher ages at KCH – but mostly within the error range), but the KCI ages on F4 are in the range of the IRH dating on T3, which then means that the closed course ends at ages where it starts although the age at KCH does not fit to those at KCS and KCI (Figure 4.13).
- The KCS ages from F1 and T3 are coherent because the tracking is started at KCS on both profiles. The same holds for the KCH ages on F1 and T1 except that only on T1 the tracking is started at KCH (which makes no difference to the former case).

The systematical mismatch of the KCH dating cannot be explained here, but the coherency of KCI and KCS is considered to represent the IRH ages even at locations near KCH. Thus, the final IRH dating is performed using only ages from KCS and KCI. If even these are diverging on several profiles (e.g. IRH no. 6), only the dating on T3 is considered because of the best coherency on this profile.

The evaluation of the IRHs finally reaches the state which has already been presented in chapter 2 (Figure 2.4): The coherency of the ice datings via GPR isochrones can be stated for 50 a in general and up to 80 a on single profiles.

IRH no.	KCS F1	KCS T3	KCI T3	KCI F4	KCI F5	KCH F1	KCH T1	IRH age
1	11.9 ± 3.2	12.1 ± 3.2	9.3 ± 3.2	13.6 ± 3.4	11.5 ± 3.4	16.9 ± 3.5	17.0 ± 3.5	11
2	15.9 ± 3.3	15.8 ± 3.3	15.9 ± 3.8	15.4 ± 3.7	12.5 ± 3.4	21.6 ± 3.4	21.0 ± 3.4	16
3	20.0 ± 3.3	19.7 ± 3.3	18.0 ± 3.8	18.3 ± 3.9	16.1 ± 3.8	23.8 ± 3.5	22.5 ± 3.5	19
4	25.4 ± 3.3	24.8 ± 3.3	23.8 ± 4.0	20.8 ± 4.0	22.6 ± 4.0	29.4 ± 3.5	28.4 ± 3.5	24
5	30.0 ± 3.3	30.3 ± 3.3	26.4 ± 4.3	23.3 ± 4.0	25.9 ± 4.2	33.4 ± 3.5	32.6 ± 3.5	29
6	35.3 ± 3.4	35.4 ± 3.4	38.3 ± 5.2	29.4 ± 4.7	30.1 ± 4.9	40.7 ± 3.5	40.1 ± 3.5	37
7	43.3 ± 3.5	43.8 ± 3.5	44.4 ± 4.0	39.7 ± 5.0	40.6 ± 4.8	48.3 ± 3.6	48.4 ± 3.5	44
8	51.5 ± 3.8	51.5 ± 3.8	46.9 ± 4.1	45.7 ± 4.1	45.3 ± 4.0	56.4 ± 4.4	54.4 ± 3.9	49
9	59.7 ± 4.2	60.1 ± 4.2	54.1 ± 4.1	45.7 ± 4.1	50.1 ± 4.1	64.7 ± 9.4	60.9 ± 8.8	-
10	66.7 ± 4.6	66.7 ± 4.6	60.3 ± 5.1	49.8 ± 4.1	49.8 ± 4.1	70.4 ± 10.1	69.6 ± 9.6	-
11	75.3 ± 5.1	75.6 ± 5.1	80.1 ± 5.8	58.0 ± 4.9	56.0 ± 4.5	81.1 ± 11.6	78.8 ± 10.7	-

Table 4.1
IRH ages from the ice core datings in years before present; the ages are obtained by evaluating the age-depth-relations of the ice cores at the corresponding IRH depths. There are 7 different ages for each IRH because each listed ice core is situated on at least two GPR profiles. The used GPR profiles are named in the header. The last column contains the estimated IRH ages used for further considerations. Note that it is not the arithmetical mean of the ages from the ice cores but the ages of KCI and KCS (and especially those on T3) which were prefered for dating the IRHs. Note further, that the three lowermost IRHs were rejected for further considerations due to their bad coherency. IRH no. 4 is marked red because it is illustrated in Figure 4.13.

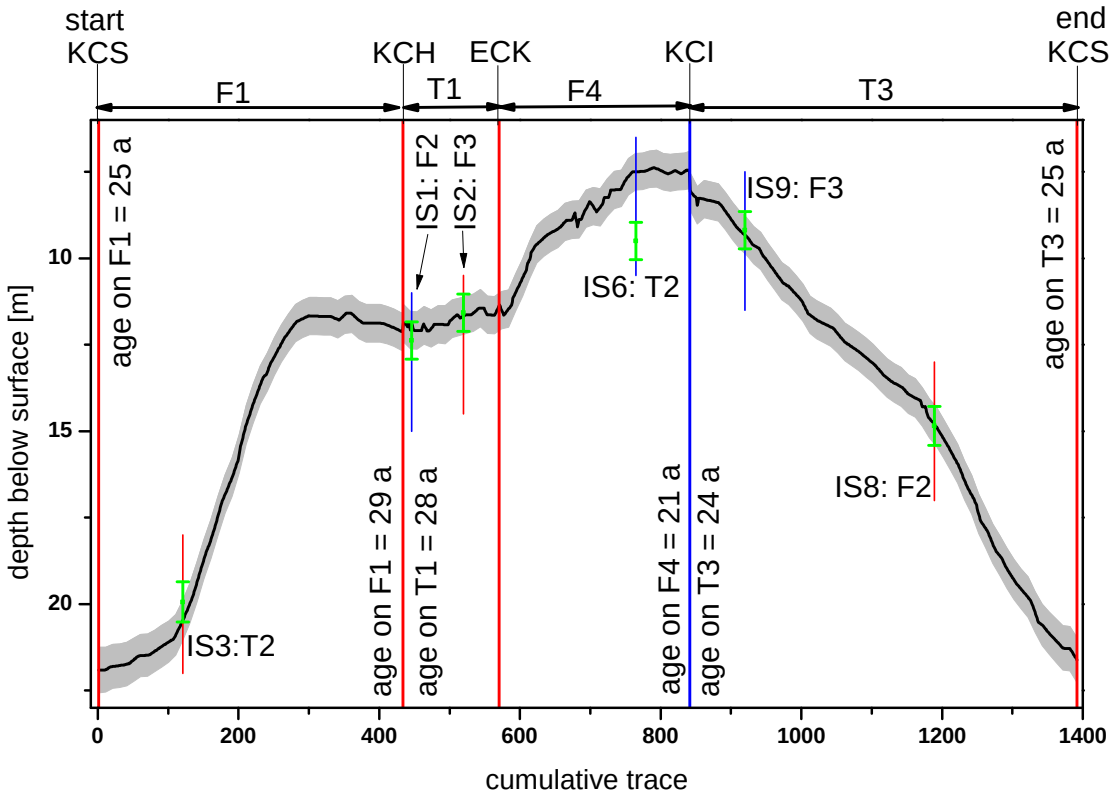
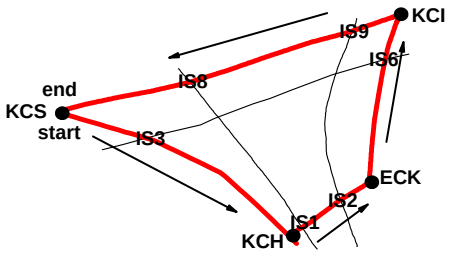


Figure 4.13
IRH no. 4 on the closed course F1 → T1 → F4 → T3; note that the T1-and F4-trace direction is reverse to the illustration in appendix B.2. The black line marks the picked and converted IRH (grey: error ∼0.6 m). The vertical lines indicate intersections of GPR profiles. Red lines: intersections, where the depths must cohere inevitably because IRH tracking was started there on at least one profile; blue lines: remaining intersections (designations as in Figure 4.8 and overview below). The green error bars stand for the depth of the same IRH on the remaining intersecting profiles (F2, F3, T2). Ice core ages of the IRH are shown in the plot as well. Note the high KCH vs. the rather low KCI ages and further the mismatch of the T2 intersections and the slight depth mismatch at KCI (within the uncertainty range). F5 is not shown in this illustration.



5 Input generation for model application to flow lines

In this chapter, the preparation of the model input parameters is discussed. In this context, the definition of the model flow lines has to be dealt with first: SYNDICATE is applied to several flow lines, especially to those defined by the GPR profiles F1-F5. Actual flow lines do not cross as F4 and F5 do at KCI. The real flow line is represented better by F5, but F4 is also considered as a flow line because the actual flow lines are not known exactly. On the GPR profiles, a comparison between modeled results, GPR isochrones and ice core datings is carried out. From now on, they are refered to as primary flow lines with the according designations of the GPR profiles (F1-F5). In addition, further (secondary) flow lines are defined via the surface topography profile (section 5.5).

At first, the input preparation for the primary flow lines is discussed, namely topography, accumulation rate, density and flux parameters. Section 5.4 deals with the spatial interpolation of the input. Finally, the definition of the secondary flow lines is presented.

5.1 Surface and bedrock altitude and accumulation rate

Surface altitude

The surface altitude is obtained from the GPS data. Measurements from 2008 and 2010 are found to be consistent at the intersections. The altitude values of F1 (2000) are about 8 m lower compared to data from 2008 which are located approximately along F1. Since there certainly is no such rough step in the surface, it is associated to GPS data acquisition and processing or potentially to actual elevation changes within these 8/10 years. Consequently, the altitudes from 2000 are rejected and taken from the surface altitude interpolation (section 5.4).

Ice thickness and glacier bed altitude

The ice thickness is obtained from the GPR data as explained in section 4.4. The bedrock altitude is determined by subtracting the ice thickness from the surface altitude.

Surface accumulation rate

The dated IRHs (in meter we) (cf. Table 4.1) are considered for the derivation of the accumulation rate. The accumulation rate along the GPR profiles can be determined by linking the layers' depth and their ages. This is done not only for the F-profiles, but also for the transversal (T) profiles for later interpolation (section 5.4). A simple division ($\dot{b} = z^{\mathrm{IRH}}/t^{\mathrm{IRH}}$) neglects annual layer thinning due to vertical strain. In order to account for this, vertical strain uniform in depth (see section 3.2) is considered. Equation (3.8) is solved for $\dot{b}$; z and H of the IRHs are inserted in meter we (cf. section 3.4). Figure 5.1 shows the accumulation rate distributions exemplarily along F1 and T3. A compilation of the accumulation rate patterns on all GPR profiles can be found in appendix C.1. The blue lines mark the accumulation rates from the uppermost IRH ($t = 11$ a). It is systematically higher than the other distributions (black). In general, the accumulation rates from older layers are lower.

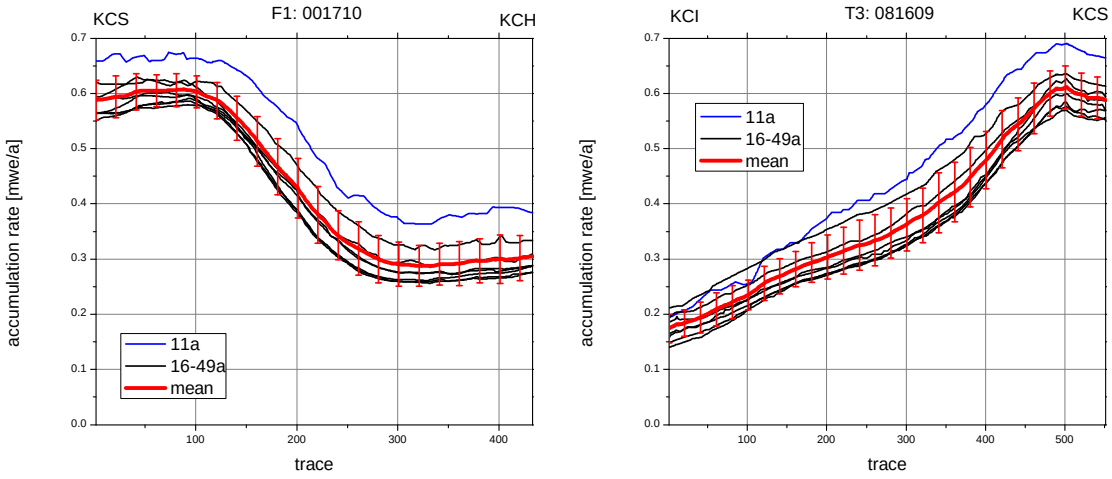


Figure 5.1
Accumulation rate distribution on F1 and T3; the blue and black lines are those calculated from the IRHs (ages are listed in the legend). The red lines are the mean rates including the blue distribution with the according error of the mean.

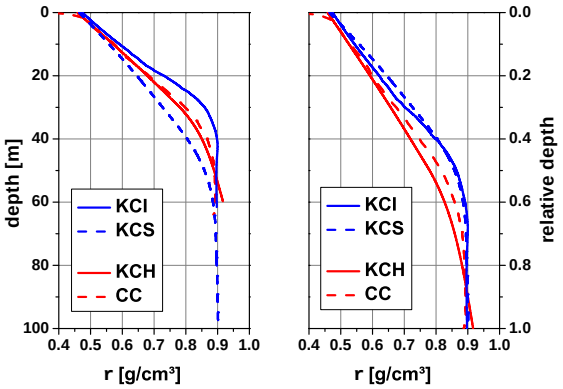


Figure 5.2
Smoothened density distributions of the four deep cores vs. depth (left) and relative depth (right); KCS- and KCI-density (located at the saddle) are similar on the Σ-axis (relative depth). KCH- and CC-density (located at the slope) are also similar – even though not as clear as in the case of the saddle cores.

5.2 Density

The specification of density input locations and of the corresponding density distributions is necessary not only as model input but also for GPR processing. In addition to the interpolation performance presented in chapter 3, a general scheme for the fixation of these has to be developed: F1 is the only flow line and the only GPR profile on which the vertical density distribution is known at two points (KCH and KCS). The four deep cores can be classified as saddle cores (KCS, KCI) and slope cores (CC, KCH). Each class features a similar smoothened density distribution with respect to the relative depth coordinate Σ (Figure 5.2). Since ECK is located at the slope, its density distribution is taken as the arithmetical mean of the two slope core density distributions at each value of Σ. The density data are taken from the database at the IUP Heidelberg. They were obtained from gravimetric (CC) and γ-attenuation [Wilhelms, 1996] (KCH, KCS, KCI) measurements. The distributions are smoothened by applying a running mean.

Starting from the density distributions $\rho(\Sigma)$ at the four locations KCH, KCS, KCI and ECK (refered to as fixed points in the following), the density at the input locations $x_{0,D1}$ and $x_{0,D2}$ of each GPR profile / flow line is interpolated according to the following scheme:

- The input locations $x_{0,D1}$ and $x_{0,D2}$ on each GPR profile / flow line are set as the intersections with the four profiles which connect the fixed points (F1, T1, F4 and T3). The intersections are illustrated in Figure 5.3 and listed in Table 5.1.

- At these intersections, the density is interpolated linearly between the two adjacent of the fixed points, i.e. for example the density distribution at intersection no. 3 is interpolated from KCH- and KCS-density with that of KCS being decisive due to the smaller distance.

To account for the smoothing, a Σ-dependent error of the smoothened density distribution $\rho_{\mathrm{sm}}(\Sigma)$ is considered:

$$\Delta\rho_{\mathrm{sm}}(\Sigma) = |\rho(\Sigma) - \rho_{\mathrm{sm}}(\Sigma)| \ .$$

In the case of the GPR processing, the error of the density interpolation is estimated via the density distribution of CC. The difference between the measured density and the according interpolation on F1 can be estimated as $\Delta\rho_{\mathrm{int}} = 0.02$ g/cm^3 (cf. appendix C.2). The interpolation error is assumed to increase with distance d from the fixed points up to this maximum error. Therefore, the error of the interpolated density distributions is

$$\Delta\rho(d,\Sigma) = \sqrt{(\Delta\rho_{\mathrm{int}}(d))^2 + (\Delta\rho_{\mathrm{sm}}(\Sigma))^2}$$

where the smoothing error $\Delta\rho_{\mathrm{sm}}(\Sigma)$ is that of the nearest of the fixed points. Finally, for each GPR profile and for each flow line the density input locations $x_{0,D1}$ and $x_{0,D2}$ and the corresponding density distributions are fixed. Consequently, the density interpolation between (and beyond) these points can be carried out as explained in section 3.5.2.

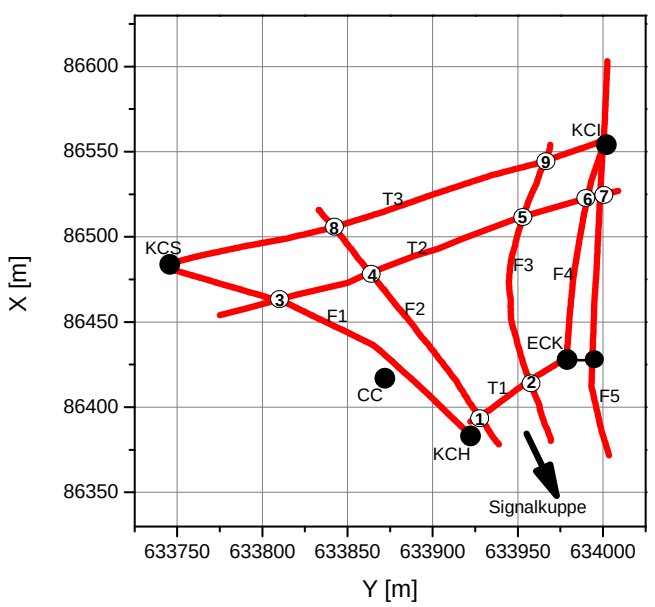


Figure 5.3
Overview of the flow lines / GPR profiles for density interpolation.

Profile	$x_{0,D1}$	$x_{0,D2}$
F1	KCH	KCS
F2	1	8
F3	2	9
F4	ECK	KCI
F5	ECK	KCI
T1	ECK	KCH
T2	6	3
T3	KCI	KCS

Table 5.1
Density input locations on the flow lines / GPR profiles; note that $x_{0,D1}$ and $x_{0,D2}$ are switched for the GPR processing on the F-profiles because the measurement was carried out by heading uphill from the saddle locations.

5.3 Flux parameters

In contrast to the previous input data which have to be handed over as a function of either the horizontal coordinate (x_0) or the vertical one, the flux parameters q_0 (determining the velocity component parallel to the surface at the upper edge of the flow line) and D (flux divergence – the fraction of ice flux that flows to the sides) are two scalar input parameters of SYNDICATE. A first approach to derive adequate values of q_0 and D is an adaptation to surface velocity measurements: According to chapter 3, SYNDICATE calculates the velocity component parallel to the surface at the surface as

$$u_S(x_0) = \underbrace{\frac{5}{4}\,\frac{\rho_{H_2O}}{\rho(x_0, Z=0)}\,\frac{1}{H(x_0)}}_{K}\left(\underbrace{(1-D)}_{A}\int_0^{x_0}\dot{b}(x_0)\,\mathrm{d}\tilde{x}_0 + \underbrace{(1-D)\cdot q_0}_{B}\right) \quad . \quad (5.1)$$

This means, A and B (and therefore D and q_0) can be determined from surface velocity measurements $u_{S,i}$ at locations $x_{0,i}$ ($i \in \{1, ..., N\}$) by solving the optimization problem[9]

$$\begin{pmatrix} K\,q_1 & K \\ \vdots & \vdots \\ K\,q_N & K \end{pmatrix} \cdot \begin{pmatrix} A \\ B \end{pmatrix} = \begin{pmatrix} u_{S,1} \\ \vdots \\ u_{S,N} \end{pmatrix} \quad (5.2)$$

where $q_i = \int_0^{x_{0,i}} \dot{b}(\tilde{x}_0)\,\mathrm{d}\tilde{x}_0$ and abbr. A, B, C defined in eq. (5.1).

Surface velocities measured by stakes were reported by Lüthi and Funk [2000] (Figure 5.4). The velocities are corrected with respect to the submergence velocity [Keck, 2001] and then projected on the corresponding flow line's tangential vector within the glacier surface plane. A further velocity vector is given at KCI by borehole location measurements in 2005, 2008 and 2010. The optimization (equation (5.2)) is carried out by Matlab using these projections. The flux parameters thus obtained are shown in Table 5.2. The surface velocity distributions resulting from the optimized flux parameters are shown in appendix C.3.

The ice slab model (in particular: equation (3.13)) suggests that $q_0 \approx \bar{\dot{b}} d_0$, where d_0 is the distance from the flow line's edge to the glacier's upper boundary (here: Bergschrund – cf. chapter 2) and $\bar{\dot{b}} \approx 0.20 \pm 0.05$ mwe/a is the mean accumulation rate in the area upstream of the flow lines' edges (estimated). In this sense, the flux parameters on F1 and F4 seem reasonable while on the other flow lines q_0

[9] $\rho(x_0, Z=0) \approx$ const; $\dot{b}(x_0)$ is already known.

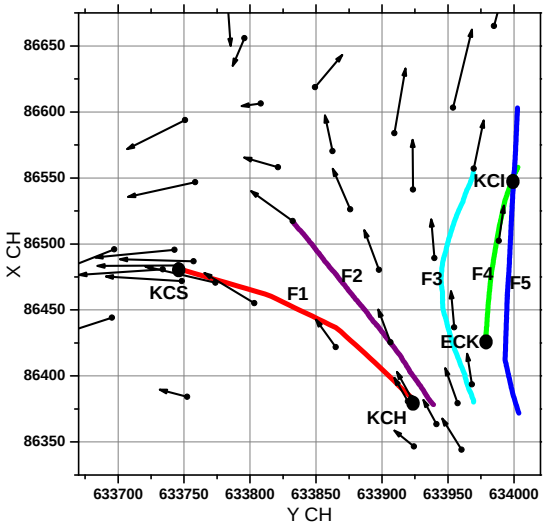


Figure 5.4
Surface velocities on Colle Gnifetti from Lüthi and Funk [2000]; the vectors are elongated by factor 30 with respect to the motion within one year. The velocities are allocated to the flow lines by eye.

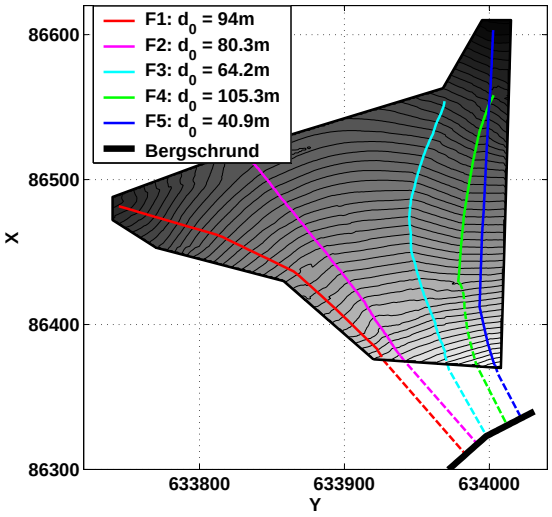


Figure 5.5
Flow line distance to Bergschrund; where the surface is given by the interpolation (section 5.4), the distance is measured along the surface altitude gradient. The Bergschrund position is estimated from plots in former theses [Keck, 2001].

flow line	q_0^{opt} [m mwe/a]	D^{opt}	d_0 [m]	q_0 [m mwe/a] from d_0
F1	21.5	0.39	94.0	18.8 ± 4.7
F2	50.7	0.71	80.3	16.1 ± 4.0
F3	47.6	0.68	64.2	12.8 ± 3.2
F4	21.9	0.42	105.3	21.1 ± 5.3
F5	0.6	0.35	40.9	8.2 ± 2.0

Table 5.2
Flux parameters from optimization and distance to the Bergschrund; d_0 is the distance of the flow lines' edges to the Bergschrund (Figure 5.5). The error of q_0 (from d_0) results from the error of the assumed mean accumulation rate (see text).

is too high (F3 and F4) or too low (F5). Furthermore, the extremely high D-values on F2 and F3 indicate that this method does not work: the surface velocity calculation of equation (5.1) does not meet the real circumstances. Consequently, this method of flux parameter calculation is partially rejected. Instead, the flux parameters are determined as follows:

- The q_0-values and -errors are taken from the distance to the Bergschrund (see above).
- $D \approx 0.40$ (from the optimization on F1 and F4 where the results seem reasonable) is assumed to be constant in the modeled area. The error of D is assumed to be 0.15:
  - It must not be chosen too low because the described way of fixing D for the model runs is not reliable.
  - Since the results for D and q_0 from the optimization on F2 and F3 are too high, the error is assumed such that the high values are beyond the uncertainty range.

5.4 Spatial interpolation

The schemes for density input and flux parameter derivation presented in 5.2 and 5.3 can easily be applied to any further flow line between the primary ones. The quantities which are needed as input to SYNDICATE along the entire flow line, namely surface altitude, bedrock topography (or ice thickness respectively) and accumulation rate (section 5.1), have to be interpolated in a two-dimensional area including all possible flow lines to which SYNDICATE is applied. For this purpose several methods are considered:

- Spatial interpolation by linear or cubic polynomials produces surfaces featuring extreme discontinuities or even fails in interpolation due to interpolation input which is quite far from being evenly spaced. This method is thus rejected.
- Thin-plate-spline interpolation performed by Matlab is also rejected because of the artefact generation from overshooting.
- Eventually, a more sophisticated geostatistical method, namely kriging, was found suitable.

In kriging, the sampled data which are subject to the interpolation are considered as a stochastically obtained data set. An introduction and discussion of the application of kriging for interpolation in this thesis can be found in appendix C.4.

The sampled data are checked for correlation (represented by the variogram – see appendix). This correlation is then used to estimate the corresponding quantities at further locations.

If the correlation between the sampled data points is parametrized adequately in the variogram, kriging does not produce artefacts in the interpolated data such as overshoots in spline interpolation or strong influence of samples at great distance as in polynomial interpolation. Via the variogram, one has direct influence on smoothing and accuracy of the interpolation. Therefore, the application of kriging implies variogram analysis with respect to the (in this case) two-dimensionality of the sample locations and a well-founded choice of variogram models which is discussed in appendix C.4 as well.

The interpolation is carried out on a rectangular grid with 1 m resolution. The illustration of the interpolation results is restricted to a polygon including the GPR profiles. The resulting interpolated surfaces are illustrated in Figures 5.6, 5.7 and 5.8. An illustration of the bedrock altitude, which is obtained by subtracting the interpolated ice thickness from the interpolated surface altitude, can be found in appendix C.4.

5.4.1 Interpolation error

By performing the kriging interpolation, the kriging error (equation (C.3)) of the interpolated quantities (designated by Z in the following) is inherently determined. However, the interpolation error is estimated in an alternative way: Several subsets of the data samples $\{(\vec{r}_i,\ Z_i)\}$ are left out (designated as $\{(\vec{r}_i^{\,o}, Z_i^o)\}$) and the interpolation is carried out with the remaining data points (refered to as $\{(\vec{r}_i^{\,no}, Z_i^{no})\}$)[10] and the variogram derived from the complete data set. In the case of ice thickness and accumulation rate, these subsets $\{(\vec{r}_i^{\,o}, Z_i^o)\}$ are the single GPR profiles. In the case of surface altitude, the subsets are defined by circles randomly generated within the area of interest.

At the omitted data points $\vec{r}_i^{\,o}$ the difference between the according quantity from the original data (Z_i^o) and from the interpolated set ($Z_{\mathrm{int}}^{no}(\vec{r}_i^{\,o})$) is considered as a function of the distance to the nearest data point ($\vec{r}_i^{\,no}, Z_i^{no}$). The interpolation error as a function of this distance is determined by linear regression. The obtained interpolation error is nonzero at the original data points $\vec{r}_i$ because of the nugget effect in the variogram (page 122). The procedure is illustrated in appendix C.4. In summary, the interpolation errors are

[10] *o* for omitted, *n.o.* for not omitted.

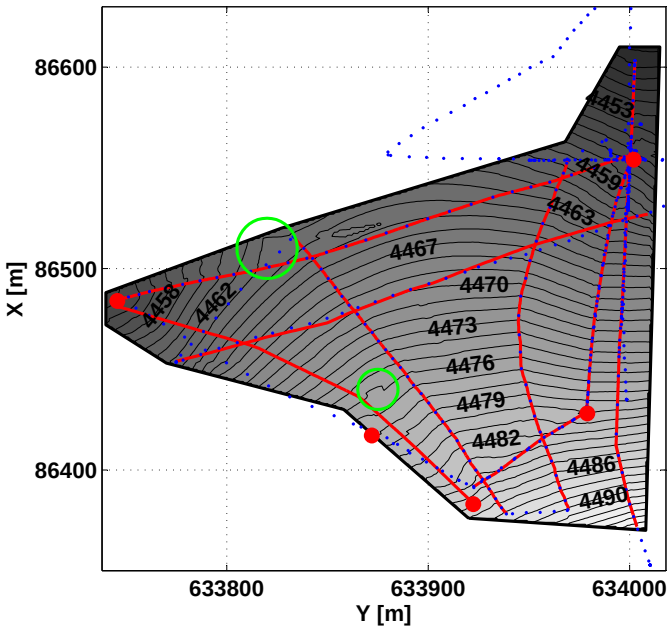


Figure 5.6
Interpolated glacier surface; the illustrated polygon is chosen because it contains all GPR profiles. Since the confidence in the interpolation decreases with the distance to the interpolation input points, an extrapolation beyond the illustrated area is not suitable.
The red lines mark the GPR profiles and the red circles represent the four deep cores and ECK. This holds for the following figures as well. The blue dots correspond to the GPS data used for interpolation (thinned). Although they cover a wider area, the interpolation illustration is restricted to the polygon mentioned above because interpolation input data of ice thickness and accumulation rate are only available along the GPR profiles.
Note that the interpolation by kriging also creates artefacts and discontinuities (green markers).

core name	H_{meas} [m]	H_{int} [m]	$\dot{b}_{\text{meas}}$ [mwe/a]	$\dot{b}_{\text{int}}$ [mwe/a]
KCH	60.3	56.9 ± 2.5	0.23	0.31 ± 0.04
CC	64.1	61.5 ± 2.8	0.22	0.29 ± 0.04
KCS	100.0	98.1 ± 2.2	0.51	0.59 ± 0.04
KCI	61.8	58.5 ± 2.5	0.14	0.17 ± 0.04

Table 5.3
Ice core depth and accumulation rate: measured values vs. interpolated values; measured values from Schäfer [1995], Armbruster [2000], Bohleber [2008].

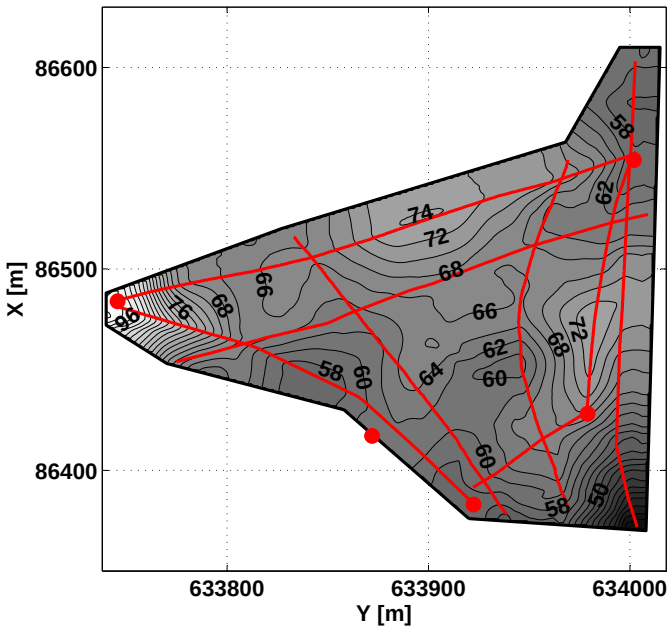


Figure 5.7
Interpolated ice thickness; the depths at the deep ice cores are systematically smaller than the actual depths (Table 5.3)

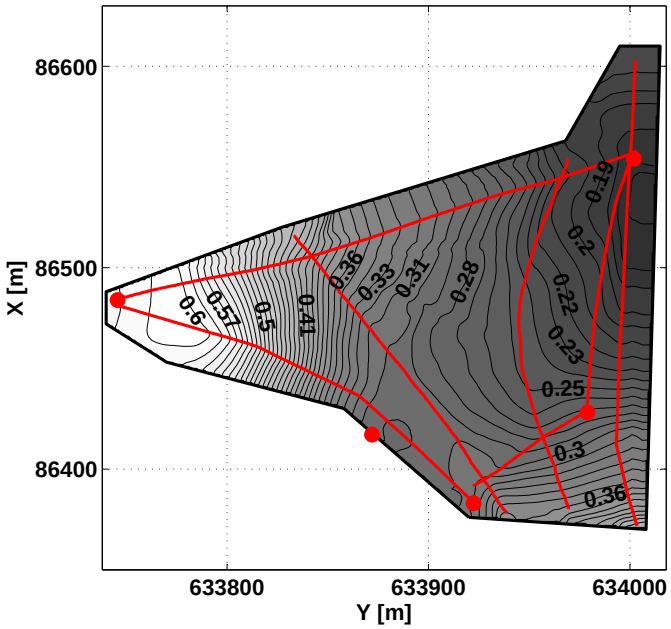


Figure 5.8
Interpolated accumulation rate by kriging (in mwe/a); note that T2 is rejected for interpolation and thus not illustrated here. The interpolated values at the deep ice cores are systematically higher than the measured values (determined by the depth of the ^{3}H-peak – Table 5.3).

- ∼1 m in the case of surface altitude. No systematic increase of the explained interpolation error can be spotted (except near KCS, where the surface altitude decreases, which cannot be reproduced by the interpolation if the data points near KCS are left out).
- ∼1-7 m in the case of ice thickness.
- ∼0.01-0.04 mwe/a in the case of accumulation rate.

The surface altitude samples, i.e. the GPS data, do not differ strongly from a general trend. Therefore, the surface variations discussed above are smaller than those of the ice thickness and the accumulation rate: The omission of data points does not imply a loss of information about local variations (in contrast to the omission of ice thickness and accumulation rate samples).

5.4.2 Validation and discussion of the interpolation

Ralph Böhlert presented interpolations of surface altitude, ice thickness and bedrock altitude based on data sets acquired in 2003 [Böhlert, 2005]. The area which he investigated is partially situated within the interpolation area as shown in Figures 5.6, 5.7, etc. He provided the original data sets from which the interpolation was carried out in his thesis. The kriging interpolation described above is applied to these as well.

The surface altitude (Figure 5.9) differs in the range of only a few meters (∼0-3 m). This discrepancy is most likely due to the GPS data acquisition and processing. Hofmann-Wellenhof et al. [2008] announce an error of vertical coordinates from GPS of at least several meters.

The ice thicknesses (Figure 5.10) from 2008 and 2010 are systematically higher than those from 2003. On T3, F3 and F4, there are differences greater than 10 m although the profiles from 2003 and from 2008/10 intersect (i.e. it is not caused by interpolation). Böhlert used a constant wave speed for the conversion of TWTs to depth. Since this neglects the firn correction, one could suspect that this might cause differences. Therefore, the originally measured TWTs are interpolated as well (Figure C.11 – appendix C.4). The difference of the TWTs features the same discrepancies as that of the ice thicknesses. Accordingly, the main reason for the mentioned discrepancies is not the difference in TWT conversion but the ambiguity in picking bedrock reflections. This might be caused by the difficulties described in section 4.5.1. Since the mean difference of the TWTs is less than 12 ns and since Böhlert lists discrepancies of ice thickness at the profiles' intersections of 0.1-6 m, this is assumed to be within the uncertainty range of the bedrock picking method.

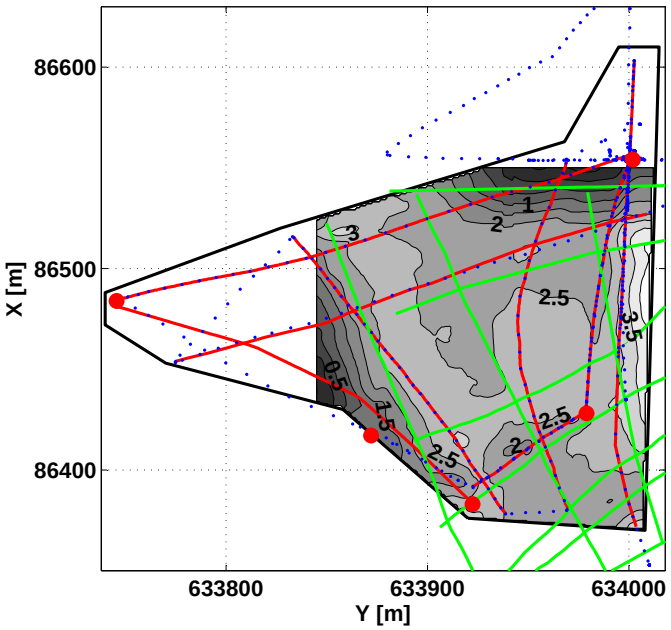


Figure 5.9
Comparison of glacier surface: 2003; red: data points from 2008/10; green: data points from 2003 [Böhlert, 2005]; the illustrated surface difference is $S^{(2008/10)}(Y,X) - S^{(2003)}(Y,X)$.

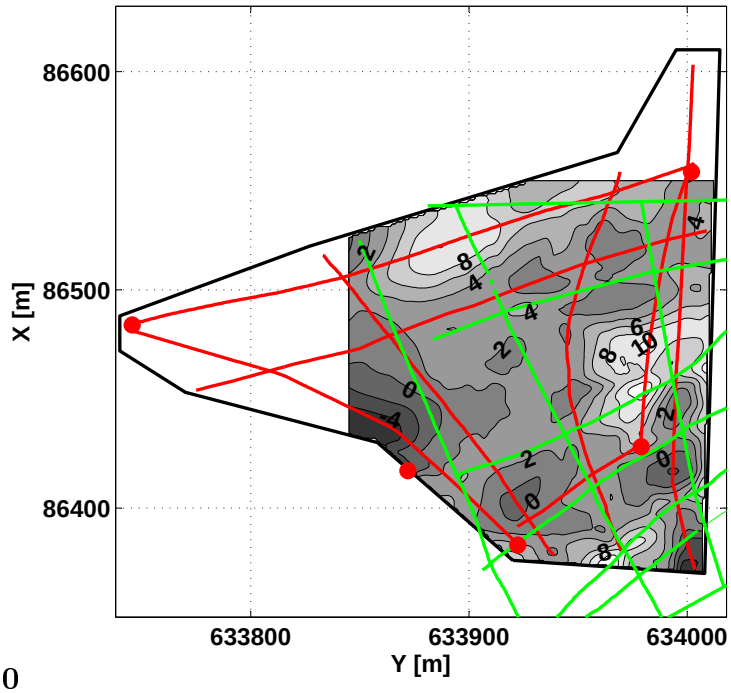


Figure 5.10
Comparison of ice thickness: 2003; red: data points from 2008/10; green: data points from 2003 [Böhlert, 2005]; the illustrated ice thickness difference is $H^{(2008/10)}(Y,X) - H^{(2003)}(Y,X)$.

5.5 Secondary flow lines

The secondary flow lines, to which SYNDICATE is applied, are principally fixed by tracking the surface altitude gradient. The primary flow lines do not fit to this gradient at every point and the surface gradient does not necessarily represent flow lines. So a compromise is made, having the secondary flow lines running between the primary ones. Since the definition of the primary flow lines is performed similarly (except that there were no other flow lines to be assumed as constraints), this method is adequate. In summary, eleven flow lines are fixed. They are illustrated in Figure 5.11.

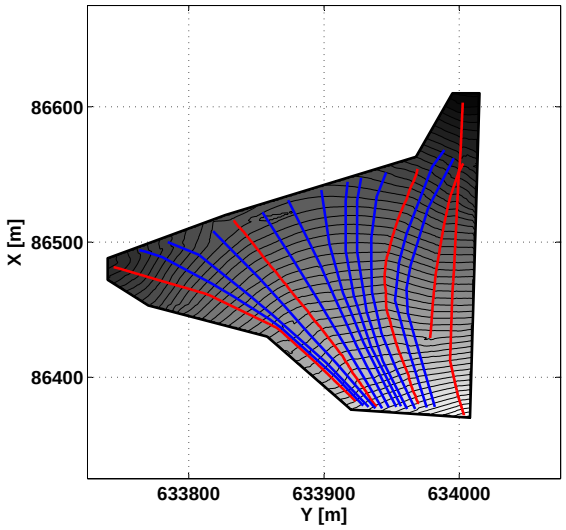


Figure 5.11
Red: primary flow lines (GPR profiles parallel to flow lines – see e.g. Figure 5.3); blue: secondary flow lines.

The input to SYNDICATE on the secondary flow lines is derived from the interpolated quantities (surface altitude, ice thickness and accumulation rate) and according to section 5.2 (density). The divergence parameter is $D = 0.40$ and q_0 is determined via the distance to Bergschrund (section 5.3). The error of the interpolated quantities (e.g. ice thickness H) consists of the derived error of the original data at the nearest point (e.g. ΔH_{GPR}) and the interpolation error (ΔH_{int}), which increases with the distance d from the original data points (except for the surface altitude, for which the interpolation error is assumed to be constant):

$$\Delta H = \sqrt{(\Delta H_{\mathrm{GPR}})^2 + (\Delta H_{\mathrm{int}}(d))^2} \ .$$

At this stage, 16 flow lines, including their corresponding input properties, are defined for model application. The model and bootstrapping runs are performed with these input data.

6 Results and discussion

In this chapter, the results from the model runs are presented and discussed. At first, the method is validated in the upper part of the glacier body where a comparison between GPR data and model results can be carried out. Then, the model results concerning ice cores (age-depth relation and source regions of particles) are treated. Finally, the information on modeled isochrones is linked to the ice cores in order to compare their age-depth profiles and to make a statement on the extension of age coherency. The chapter concludes with an overall assessment of the method.

6.1 Comparison of GPR and modeled isochrones

The model's capability to predict isochrones in agreement with IRHs from GPR needs to be validated. Thus, the isochrones from GPR and from the model runs on the primary flow lines are compared. Figures 6.1 and 6.2 exemplarily illustrate the comparison of isochrones from SYNDICATE and GPR along the flow lines F1 and F3. The comparison along F2, F4 and F5 can be found in appendix D.1.

The mean error of the GPR isochrones is about 0.5-0.8 m (cf. section 4.5.1). The uncertainty of the modeled isochrones is determined by the bootstrap introduced in section 3.5.4. In the case of the compared ones (i.e. those of ages $t \leq 49$ a), it varies between 0.9 m up to 2.6 m increasing with depth.

Discussion of the GPR-model comparison

On F1, there is a systematic mismatch of the modeled isochrones which is most likely due to shortcomings in the SYNDICATE's output ages and to the steep bedrock topography (Figure 6.12). In contrast, the modeled isochrones on F3 reproduce the GPR isochrones very well. In general, the agreement of isochrones is adequate with respect to the shape: In most cases a shift in depth and/or in horizontal direction (x_0) leads to better agreement (Figure 6.3). As already mentioned above, a potential mismatch in depth is probably due to SYNDICATE's poor dating ability.

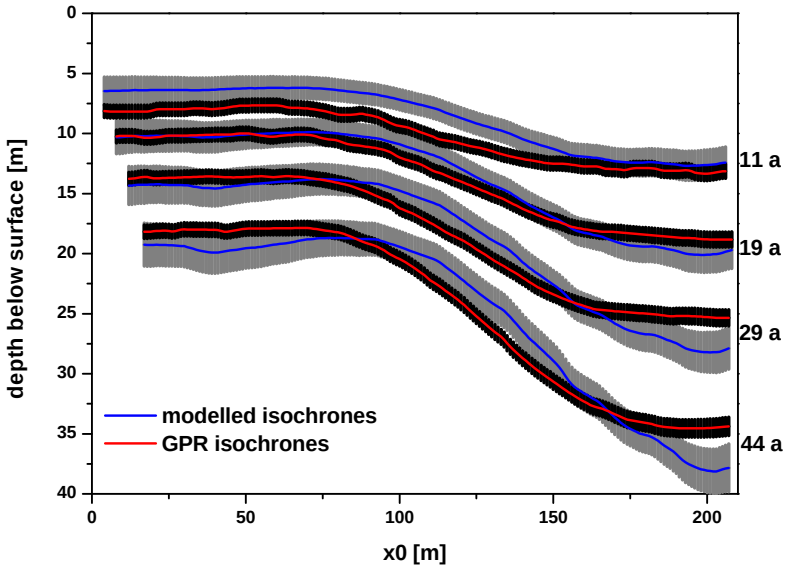


Figure 6.1
Comparison of selected isochrones from GPR and SYNDICATE along profile F1; the grey bonds indicate the error of the modeled isochrones, the black ones those of the picked GPR-IRHs. The ages of the illustrated IRHs and model isochrones are on the right. On F1 the agreement of GPR and model is worse in upper parts (isochrone $t = 11$ a) and at large x_0.

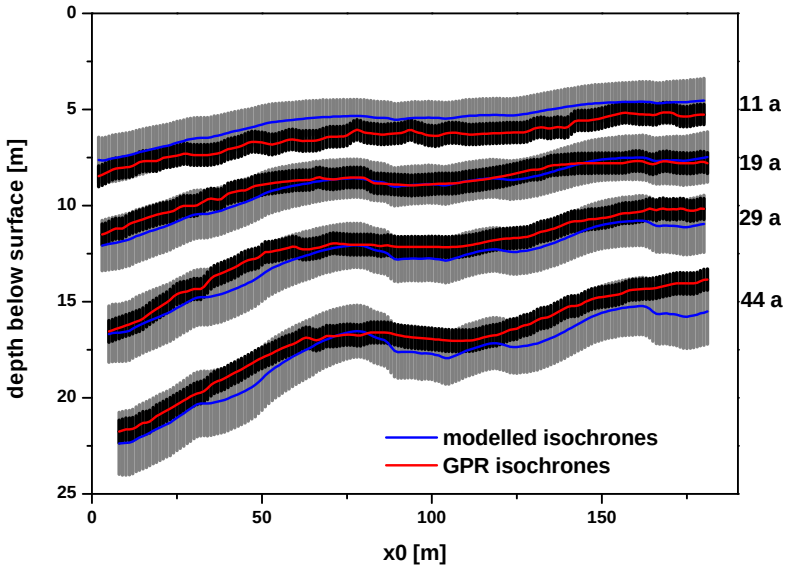


Figure 6.2
Comparison of selected isochrones from GPR and SYNDICATE along profile F3; the grey bonds indicate the error of the modeled isochrones, the black ones those of IRHs. The ages of the illustrated IRHs and model isochrones are on the right. There is good agreement between GPR and model.

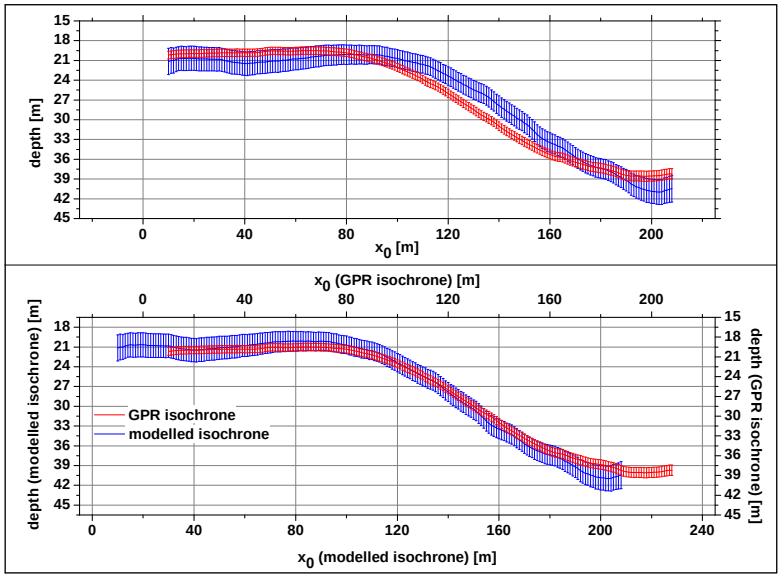


Figure 6.3
Top: modeled and GPR isochrone ($t = 49$ a) along F1; bottom: manual offset correction of modeled and GPR isochrone for higher agreement; the horizontal shift is $\Delta x_0 = 20$ m and the vertical shift is $\Delta Z = 1.5$ m. The shifted isochrones then agree well. Note that the offset is only carried out in this plot in order to illustrate the agreement of isochrones in shape.

The agreement in shape can be interpreted as a validation of the method: SYNDICATE provides isochrones which exhibit the same structure as those derived from GPR processing. Consequently, there is reason to assume that the modeled isochrones beyond the IRH reach meet the structure of the actual age distribution.

6.2 Ice core related results

Can SYNDICATE be used for ice core dating?

Although it is not named as objective, the question arises whether the vertical age distributions which are output of SYNDICATE are use- or helpful for ice core dating and whether the separate age information obtained from SYNDICATE is reliable. The answer to this question is found by comparing the KCS and the KCI datings obtained from experiment as described in chapter 2 to those obtained from SYNDICATE by application to flow lines F1 (KCS) and F4 and F5 (KCI). These comparisons can be seen in Figures 6.4 and 6.5. Since the dating mismatch is the depth-integral over the mismatch in vertical velocity, the latter one is illustrated as well.

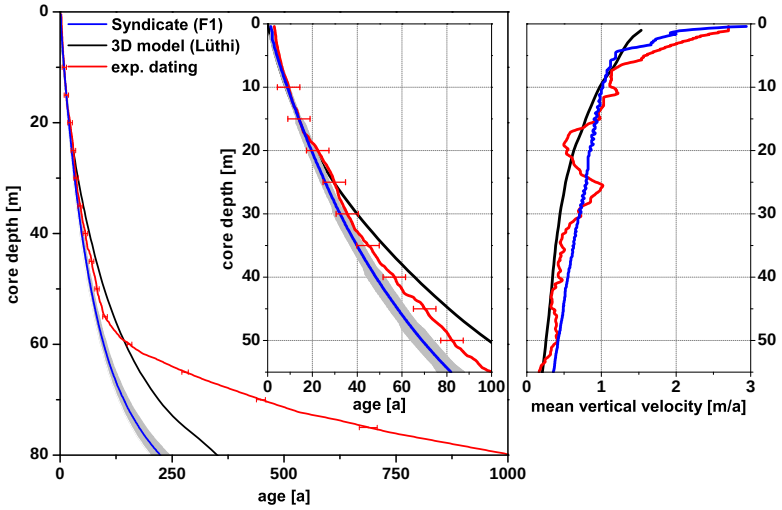


Figure 6.4
Left: comparison of experimental (see chaper 2) and modeled dating at KCS including the dating from Lüthi and Funk [2000]; the additional frame shows a zoom of the upper region. The grey bond marks the error of the SYNDICATE dating from bootstrapping (section 3.5.4). Right: vertical velocity at KCS for the same data sets.

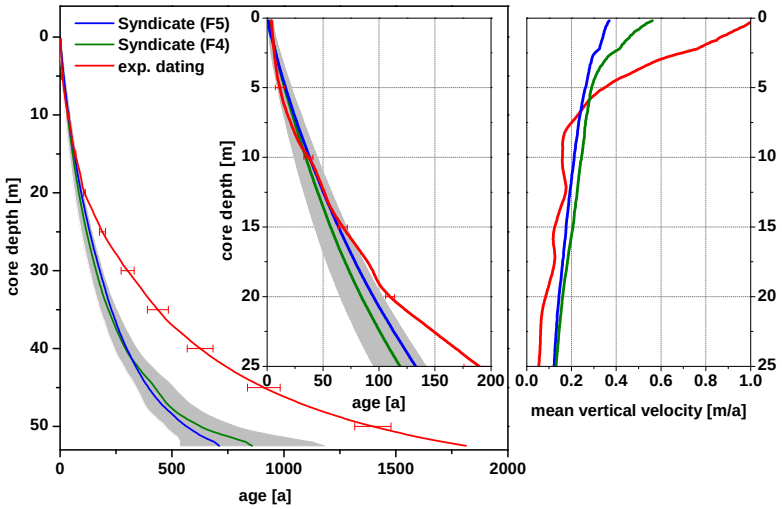


Figure 6.5
Left: comparison of experimental (see chaper 2) and modeled dating at KCI; the additional frame shows a zoom of the upper region. The grey bond marks the error of the SYNDICATE dating from bootstrapping (section 3.5.4) on F4. Right: vertical velocity at KCI for the same data sets.

KCS
As a further comparison, the modeled dating according to the three-dimensional modeling on Colle Gnifetti by Lüthi and Funk [2000] is shown.[11] The following is found:

- The characteristic bend of the experimental dating at a core depth of ∼57 m is neither reproduced by SYNDICATE nor by the 3D model.
- The discrepancy between SYNDICATE and the experimental dating already occurs above this bend (no agreement within the uncertainty range of the two data sets).
- The vertical velocity (derivative of the age-depth profile) shows a better agreement than the age-depth relations: The experimental one oscillates about that from SYNDICATE in the zoomed region (lack of smoothness due to the illustration as a function of depth in meter instead of meter we).
- The result of SYNDICATE does not fit worse to the empirical data than that from the 3D model.

KCI

- The experimental dating features a similar bend as KCS at ∼19 m. Up to this bend, the experimental and the modeled dating fit within the error range.
- The same mismatch between model output and ice core dating as at KCS occurs below the bend.
- The agreement of vertical velocities is not as good as at KCS: Below the intersection at ∼6 m core depth, the experimental one is systematically lower that the two modeled ones.
- The results from SYNDICATE on F4 and F5 fit to each other.

A further approach is made by testing whether the agreement of experimental ice core dating and model output can be improved by adjusting the model input: The accumulation rate along the flow line F1 is varied such that the KCS model dating fits to the experimental one, i.e. the bend occurs in the model dating.

The accumulation rate is assumed to increase from $\dot{b} = 0.05$ mwe/a up to $\dot{b} = 0.60$ mwe/a. The horizontal coordinates x_0 at which each $\dot{b}$-increasing step is

[11] Note that the illustrated model dating by Lüthi and Funk might not be the final dating result at KCS but a preliminary one which was also used by Armbruster [2000] for comparison.

carried out are varied manually until the output dating is in good agreement with the experimental one. The remaining input parameters are chosen according to chapter 5. The result of this procedure is shown in Figure 6.6

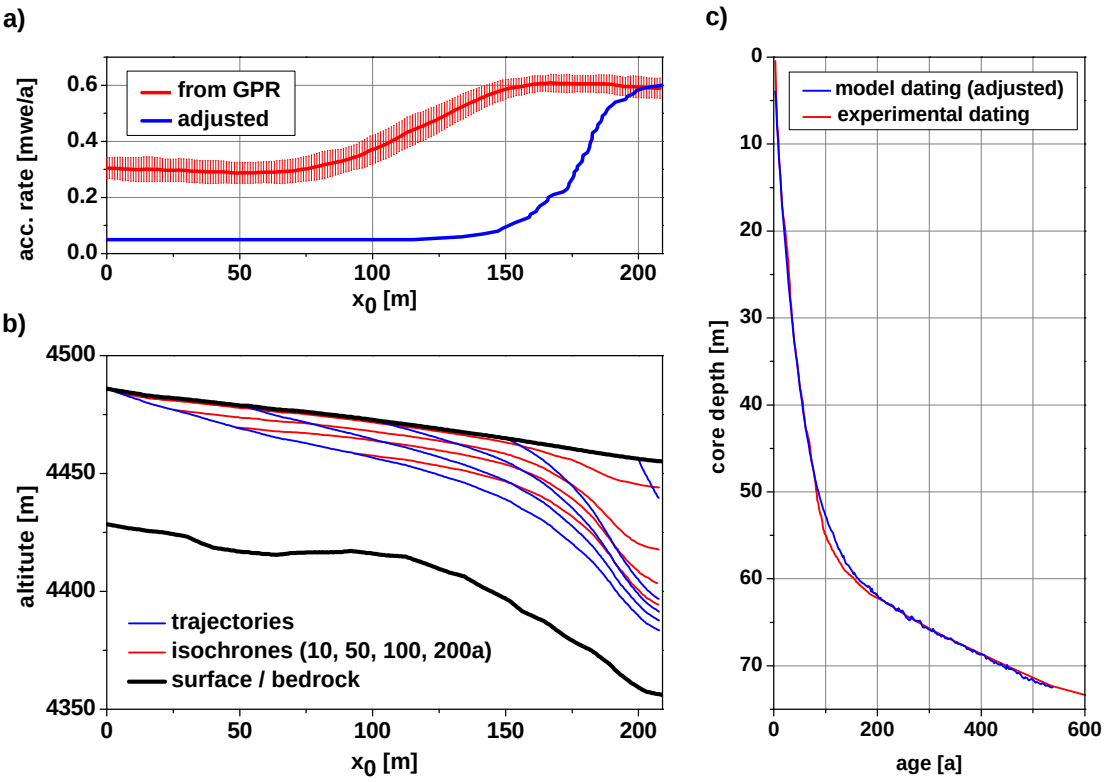


Figure 6.6
a) Accumulation rate from adjustment and from the input derivation according to chapter 5; b) exemplary isochrones and trajectories from adjusted accumulation rate application; c) experimental and modeled KCS dating; the agreement of these two datings is subject to the adjustment. The resulting accumulation rate is unreasonable (see text).

Discussion of SYNDICATE's dating ability

In the context of age-depth profiles obtained from modeling, one first has to admit that the experimental datings have been used in input generation for SYNDICATE: The IRH dating is dominated by the age-depth relation at KCI and KCS (section 4.5.2). Since the dated IRHs are essential for the accumulation rate calculation, SYNDICATE's output presented in Figures 6.4 and 6.5 is not independent from the experimental dating (in contrast to e.g. the 3D model).

In the presented cases, the agreement of the age-depth relations can only be improved by applying input parameters obtained without physical basis which then lead to inappropriate results (Figure 6.6): The two presented accumulation rate distributions feature the same general course along F1 but the difference in accumulation rate between saddle and slope locations is unreasonably high. Furthermore, the transition from low to high accumulation rates takes place too close to the KCS location and within a too small distance. In principle, such transitions are possible. They occur in company with elevation changes and the related prevention of wind erosion on the lee side. However, there are no such prominent elevations on Colle Gnifetti. This leads to the conclusion that the adjusted accumulation rate distribution lacks a physical basis.

Consequently, the ages of SYNDICATE are unreliable. Since the original objective is to derive the isochrone structure by using SYNDICATE and not necessarily to date ice cores, this insight does not imply the rejection of the method – especially because the relative situation of SYNDICATE's ouput, i.e. the isochrones' courses along a flow line, is adequate (section 6.1).

6.2.1 Source regions of particles in the ice cores

One of the flow modeling's objectives related to ice cores are the source regions (at the glacier surface) of the particles in the ice core. It has already been outlined in the introduction and in chapter 3 that SYNDICATE will be used for this purpose by rearranging the trajectories with respect to their depth at the drilling site. The dating of an ice core is directly related to annual layer thickness and the annual layer thickness depends on the accumulation rate at that site where the corresponding particles have been accumulated on the glacier. Therefore, the results from this examination are expected to improve the interpretation of ice core properties.

An according study has already been carried out for KCS [Lüthi, 2000]. In the case of KCI, this study is the first one related to source regions. Only these two cores are considered here because SYNDICATE needs input far enough upstream of the examined drilling site which is not given in the case of KCH and CC. In Figure 6.7, the results and their possible usage are illustrated in the case of KCI. An overview over the results concerning KCS is given in appendix D.2. Note that the accumulation locations obtained by SYNDICATE are naturally linked to the trajectory structure instead of being linked to the isochrone structure. Therefore, they mainly depend on the flux parameters (cf. Table 3.2) and can be considered independently from the isochrones.

The bad agreement of the two source region distributions from the model runs on F4 and F5 are most likely due to the consideration of both F4 and F5 as flow lines towards KCS or to an inadequate flux parameter choice (or both).

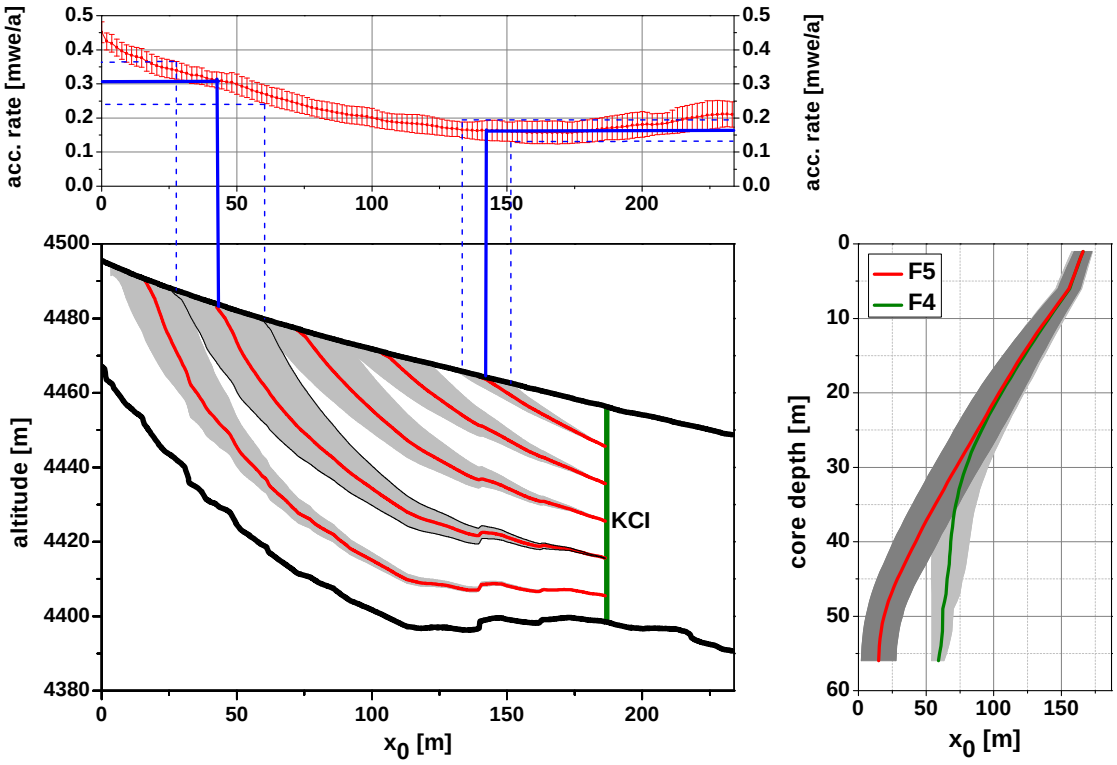


Figure 6.7
Left bottom: exemplary trajectories (red) and uncertainty (grey) on F5 leading to specific depth at KCI; left top: accumulation rate pattern along F5 (chapter 5); a possible usage of the data set is illustrated by the blue lines (dashed lines: corresponding uncertainty): The layer in a specific core depth can be allocated to the region where it has been deposited. At the corresponding x_0-coordinate, the accumulation rate can be evaluated and therefore be allocated to the core depth, where the consideration started. Right: complete distribution of core depth vs. accumulation location on F4 and F5. Note that the distribution derived from the model run on F4 is shifted with respect to x_0 in order to make the two distributions comparable. Note further that the two distribution do not agree in the lower part.

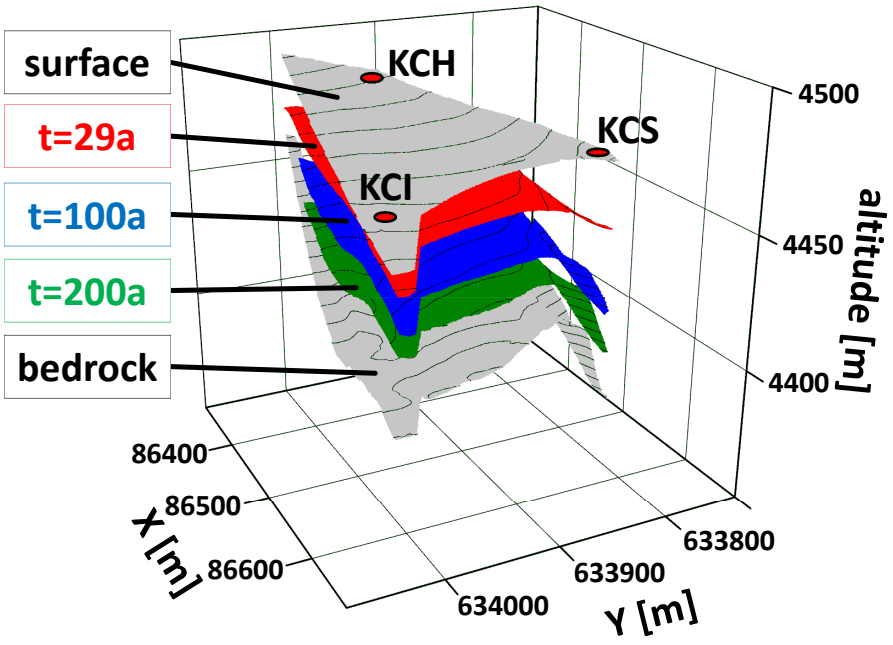


Figure 6.8
Exemplary isochronous surfaces within the firn body; the ice core locations are estimated.

6.3 Three-dimensional age distribution

The results presented in the former sections are all obtained by the application of SYNDICATE to the primary flow lines. The qualitatively adequate situation of the modeled isochrones compared to those from GPR is to be extended in lateral dimension at this stage by taking into account the results of the model runs on the secondary flow lines. The modeled isochrones of each age are sampled on the 16 flow lines and the corresponding isochronous surfaces within the glacier are interpolated by kriging as discussed in section 5.4 and appendix C.4. The results for exemplary ages are shown in Figure 6.8. Since the isochronous surfaces do not feature unreasonable discontinuities or artefacts (see below for one exception), their structure can be assumed reasonable at this stage although the modeled ages alone are unreliable (section 6.2).

6.3.1 Age allocation to model isochrones by ice core datings

The actual ages of the interpolated isochronous surfaces are determined by evaluating the datings of KCI and KCS at the corresponding depth of these surfaces

(age range: t = 40-150 a with an increment of 10 a). The age-depth relations of CC and KCH are left out in these considerations because they do not lie within the area where data points from model runs exist (see below and appendix D.5).

At first, the interpolated surfaces are considered along T3 (Figure 6.9). The artefacts near KCI in the top plot of Figure 6.9, suggest that the data points from which the surfaces are interpolated are inconsistent at KCI.

The results on the single flow lines have thus been checked for inconsistency. The isochrones on F5 reveal that the combination of the steep surface at low x_0 and the relatively high accumulation rates obtained from GPR results in numerical instabilities of the model. Consequently, the isochrones oscillate in an arbitrary manner and affect the spatial interpolation of the isochronous surfaces such that the illustrated peaks on T3 appear (illustrated in appendix D.3). This problem is circumvented by applying SYNDICATE to a shortened version of F5 starting 40 m downhill from the original starting point. The resulting isochrones on T3 then are appropriate (bottom plot in Figure 6.9). In the following, the data including F5 in full length is refered to as set 1 and those including the shortened version of F5 is refered to as set 2.

The depths of each of the modeled and then interpolated isochrones at KCI and KCS lead to the corresponding ages of these two cores. If the modeled isochrone is an actual isochrone, the ages of KCI and KCS in the corresponding depths must be the same (regardless of the model age) within the error range. This is a necessary condition, but not a sufficient one. The ages of the specific modeled isochrones at KCI and KCS are shown in Figure 6.10: The KCS age is plotted vs. the KCI age. Again, it is clear that the modeled ages do not correspond to those from the ice core datings. If the ages of the two ice cores are the same on the isochrones, the data points are situated on the bisecting line.

The errors (both on abscissa and ordinate) are composite of the dating error (Δt_{dat}) and the depth error of the isochrones obtained from the bootstrap (Δz):

$$\Delta t = \sqrt{(\Delta t_{\mathrm{dat}})^2 + \left(\frac{\mathrm{d}t_{\mathrm{dat}}}{\mathrm{d}z}\,\Delta z\right)^2}\ .$$

There is a trend towards KCS ages which are too high compared to the KCI ages. Due to lower KCI ages in the case of set 2, the trend is more prominent in the corresponding data (right plot in Figure 6.10). However, the results from set 1 are inconsistent at KCI and thus have to be rejected when considering the dating coherency.

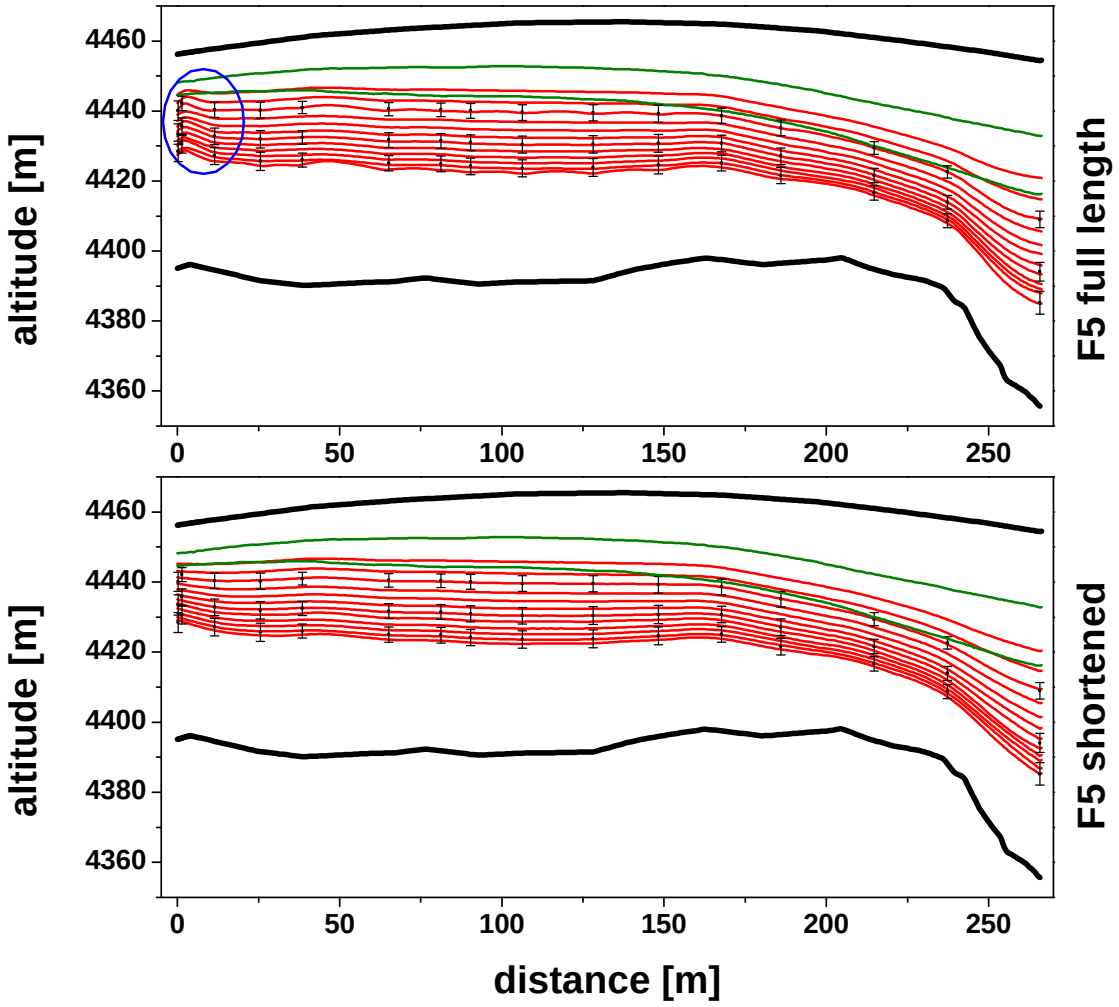


Figure 6.9
Interpolated isochrones on T3; in both plots, KCI is on the left (0 m) and KCS is on the right (266.4 m). The error bars result directly from the bootstrap (section 3.5.4) and are therefore not necessarily situated on the illustrated interpolated isochrones. Top: interpolated from the model run with F5 in full length (set 1 – see text); note the unreasonable peaks (blue mark) near KCI. Bottom: interpolated from the model runs with F5 shortened (set 2). The corresponding variograms are shown in appendix D.4. The internal reflection horizons of ages $t = 24$ a and $t = 49$ a are plotted in green for comparison.

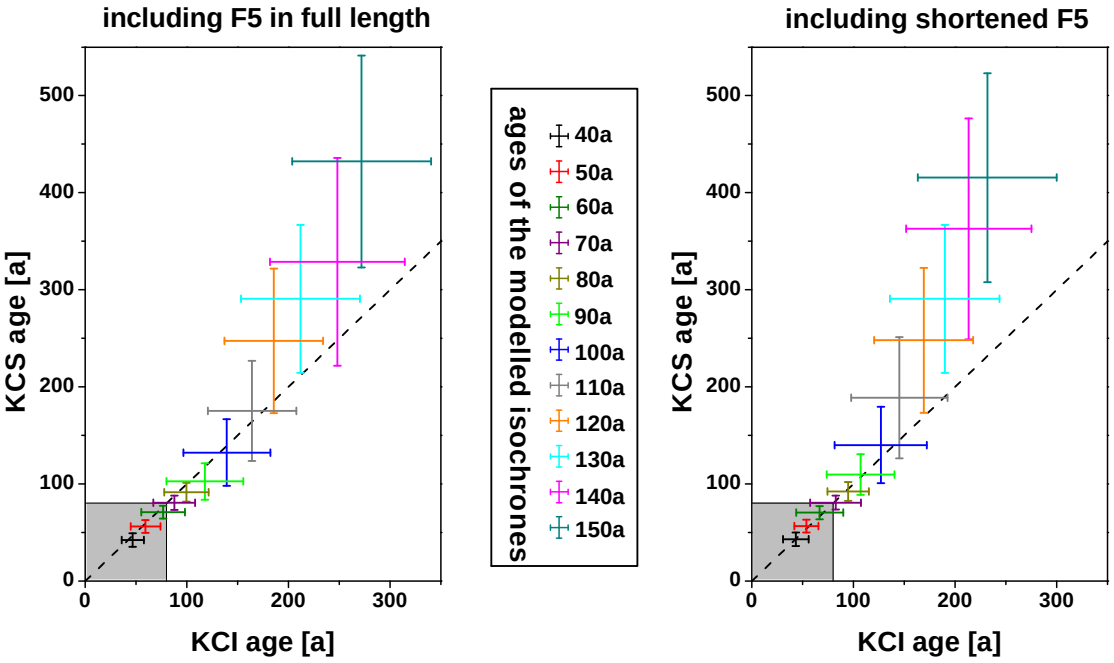


Figure 6.10
KCS age vs. KCI age of modeled isochrones; each data point corresponds to a modeled isochrone indicated by color (see legend in center). The grey shades mark the age range in which the KCI-KCS-coherency can be found via IRHs only (Table 4.1). Left: ages according to the interpolation from set 1; right: analogous from set 2.

In the following, only the results from set 2 are considered. At $t = 110$ a (model age), which corresponds to core ages of ∼170 a, the mentioned trend becomes dominant. However, the high uncertainties imply a possible dating coherency up to isochrones of model ages of $t = 140$ a (≙200 a KCI age) between KCI and KCS. Additionally, it can be noted that thereby the dating of the modeled isochrones by the experimental ice core datings works up to these ages.

6.3.2 Dating coherency of the four deep cores

After having examined the coherency of KCS and KCI ages on the modeled isochrones, the final step is to compare the ages of all four deep cores, i.e. to include the slope cores KCH and CC in an analogous consideration as in section 6.3.1. This comparison is carried out by evaluating the dating of these two cores at the corresponding depths of the model isochrones as well.

The age comparison yields an unexpected result (Figure 6.11): The ages of the

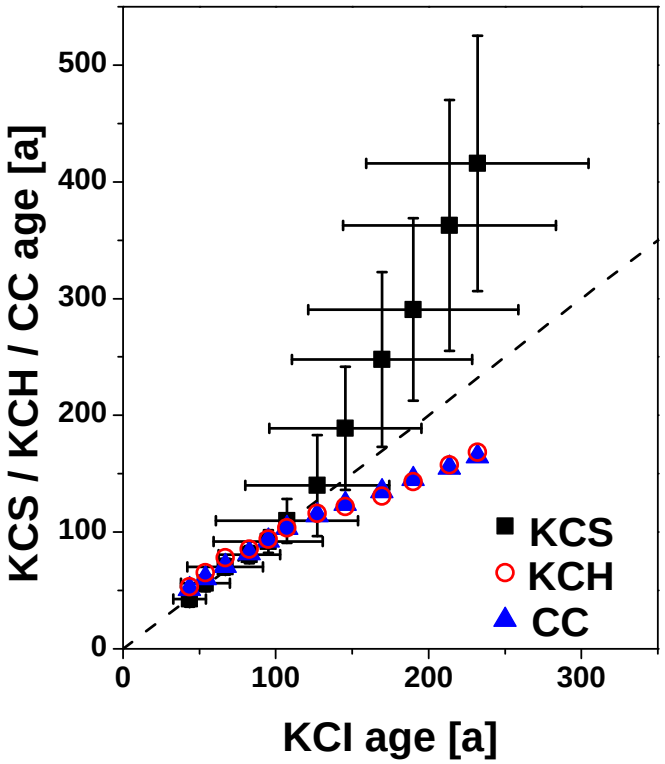


Figure 6.11
Coherency of the ages from the four cores obtained from modeled isochrones; KCS, KCH and CC age are plotted vs. KCI age. This is arbitrarily chosen as reference. The data points from KCH and CC are plotted without error bars because these can be derived solely from the dating error (there are no depth errors available from the bootstrap as in the case of KCI and KCS, but these are expected to be even more dominant at KCH and CC due to the higher interpolation uncertainty). Additionally, the results are listed in Table 6.1.

interpolated isochronous surfaces at KCH and CC are nearly the same, even in the regions where KCI and KCS do not cohere any more via the model isochrones. This is unexpected because the distance of isochronous data points for interpolation to the KCH location increases with increasing model age (shown in appendix D.5 for $t = 60, 100, 140$ a). Consequently, the interpolated surfaces become more and more unreliable at KCH. The good agreement of CC and KCH might then be due to the rather young ages of the modeled isochrones according to the CC and KCH dating.

In contrast to that, the ages of the two saddle cores do not fit to those from the slope cores beyond a KCI age of $t \approx 130$ a. KCS tends to higher ages than KCI and the slope cores tend to lower ages. KCI seems to fit better to the slope cores

than to KCS at higher ages. This is not surprising because the ice at higher ages stems from slope locations. The discrepancy of KCI / KCS and the slope cores' ages starts at ages where in the single datings of the two saddle cores (KCI / KCS), the characteristic bend (Figures 6.4 and 6.5) is passed, which might indicate that there is a connection.

Finally, the coherency of the four cores' datings can be extended up to ∼120 years before present by the isochrones structure derived from the presented method. Between single sets of cores, the method even proves a coherency up to ∼200 a with an increased uncertainty of 60-80 a.

model age	KCI age	Δ KCI	KCS age	Δ KCS	KCH age	CC age
40	44	11	43	7	53	51
50	54	16	56	7	65	61
60	67	25	70	7	78	71
70	83	20	81	7	85	82
80	95	36	92	10	94	93
90	107	47	110	19	104	103
100	127	47	140	43	116	115
110	145	50	189	53	122	124
120	169	59	248	75	131	135
130	190	69	291	78	143	146
140	214	70	363	107	157	155
150	232	73	416	109	168	165

Table 6.1
Ice core ages of the modeled isochrones; the listed ages (in years before present) are those illustrated in Figure 6.11. Δ means the errors of the listed datings in the case of KCI and KCS as described in the text.

6.4 Critical evaluation of the method

Shortcomings in input generation

Unreliable input data are mainly those of the bedrock topography. The uncertainty from GPR processing in bedrock reflection picking (section 4.5.1) and therefore in ice thickness are probably not fully represented by the error calculation (appendix B.3) and within bootstrap (appendix A.4): Eisen et al. [2003] state an ice thickness uncertainty of 5 m. Moran et al. [2000] list ice thickness uncertainties up to 36% in unmigrated data and 16% in migrated data (2D-migration). This is in clear contrast to the estimation in this thesis (∼2 m resulting from the assumed uncer-

tainty of the picked TWTs of bedrock reflection: $\Delta t_{\mathrm{TWT}} = 20$ ns). This can be seen in the comparison of ice thickness interpolated from the GPR profiles which are treated in this thesis to that from 2003 [Böhlert, 2005].

The GPR profiles which are evaluated within the scope of this thesis have been recorded within a time interval of ten years. With respect to the steady state assumption underlying the model, this should not be problematic, but short time variations of e.g. accumulation rate then lead to inconsistency.

In general, the accumulation rate input features inconsistencies in derivation. The IRH dating suffers from the systematically higher ages at KCH which results in the consideration of only the KCI and KCS ages for this purpose (section 4.5.2). This consequence implies that the KCI and the KCS ages are assumed to be more reliable in some sense. Furthermore, the higher accumulation rates from younger horizons (section 5.1) indicate either that the assumption of constant vertical strain, which is applied for calculation, is not sufficient or that the strong variations cause this systematic effect. A more adequate consideration might then be to calculate the accumulation rate rather from older horizons because of their integral and therefore averaging nature.

Shortcomings in modeling

The shortcomings of SYNDICATE lie in the fundamental simplifications. They restrict the model with respect to the output ages (which are not object of interest) and with respect to general output in deeper regions of the glacier (which is object of interest: a further extension of the coherency of the ice core datings needs an adequate treatment in greater depth):

1. **3D problem vs. 2.5D treatment:** The parametrization of transversal ice flux cannot account for all aspects of the problem's three-dimensionality. Especially the bedrock topography's influence on the flow behaviour is underestimated by this. As it is discussed in chapter 3, the kinematic condition means that the flow velocity close to the bedrock is parallel to it. Two-dimensional modeling can only account for the bedrock altitude gradient in one direction. Figure 6.12 illustrates several sites where the two-dimensional gradient and the flow lines are far from running parallel.
   A more sophisticated version of SYNDICATE has been developed which treats the divergence parameter D as a function of the distance along the flow line (x_0) and thus makes the model more adequate with respect to the substitution of the third dimension by D. However, this version has been rejected due to the lack of information about how to derive possible divergence distributions.

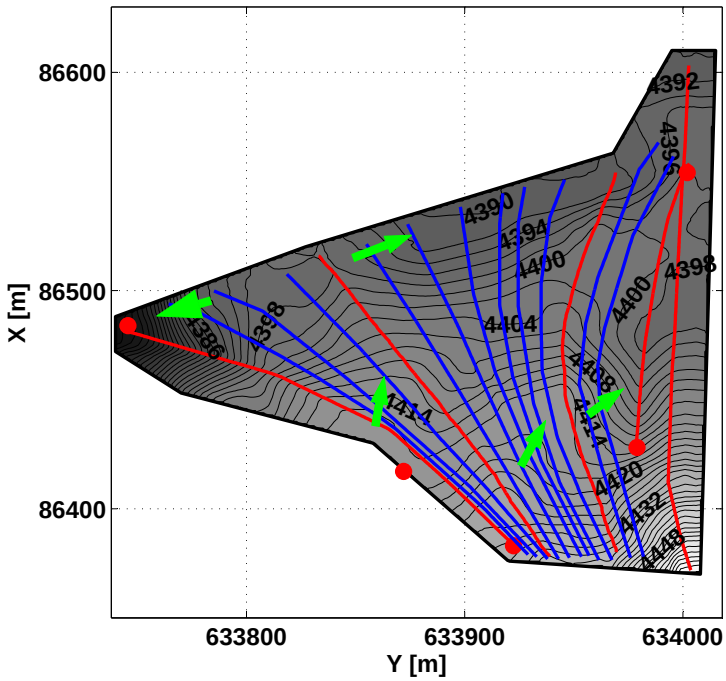


Figure 6.12
Bedrock topography and flow lines; red: primary flow lines; blue: secondary flow lines; green arrows mark regions where the bedrock altitude gradient strongly differs from the flow line direction. At these locations, the situation is expected to be hardly representable by two-dimensional modeling near bedrock. This might cause the modeled isochrones to reach too great depth near KCS which then end up at too old KCS ages of 3D isochrones (Figure 6.11).

2. **Flow law parameters:** The parameters in the flow law of ice (equation (3.5)) are chosen to be constant. This is simplifying for several reasons:
   - The exponent n does not necessarily equal 3, but depends on the stress state of the material [Paterson, 1994].
   - The flow parameter A strongly depends on temperature (amongst others) and is therefore surely not constant.
   - A change of the rheology might occur in greater depth (below the firn-ice transition). This might also be the cause for the bends in the age-depth relations of KCI and KCS and for the extremely inadequate accumulation input to SYNDICATE, which is necessary for the bend's reproduction by the model (Figure 6.6).

3. **Firn fraction:** Accounting for firn compaction by simply scaling velocities

with density instead of applying a more adequate flow law for firn is most likely one of the main causes for the poor dating ability at even rather small depths. With respect to the isochrone structure, it is appropriate: This is proved by the agreement of model isochrones and IRHs.

4. **Steady state:** Steady state in the case of the surface topography might be considered as given for the past (chapter 2), but the accumulation rate is subject to variations. This cannot be reflected by SYNDICATE and might cause further discrepancies in comparison with empirical data.

Combination

Despite the named shortcomings, the combination of GPR and SYNDICATE has been carried out successfully. The restriction of the success to a certain upper fraction of the glacier thickness is a feature which can only be circumvented by a much more sophisticated approach, e.g. a three-dimensional one [Lüthi, 2000]. However, the singularity at the ice-bedrock interface might still cause restrictions. In this sense, the derivation of a three-dimensional age distribution which is assumed reasonable for the first 200 years is a result which justifies the choice of this rather simple method.

7 Conclusion and Outlook

Assessment of the results with respect to the objectives

The original objectives of this thesis could be reached: Both a three-dimensional age distribution which is reasonable for about 0-200 years before present and an estimation of source regions of the ice deposited in the ice cores has been obtained by the combination of GPR profiles, flow modeling and the ice core data.

The structural reproduction of GPR isochrones by SYNDICATE and the dating coherency of the ice cores on the modeled isochrones give rise to the conclusion that the age distribution obtained by model isochrone interpolation reflects the real conditions quite well. Nevertheless, the ages of the isochrones themselves could not be obtained without falling back on the ice core datings again.

Since the revision of ice core datings and the interpretation of climate related records are not subject to this thesis, the source region distribution for KCI and KCS and their significance has not yet been assessed. A statement whether they can support examinations of the records has to be done in the future.

Recommendations for further studies

A two hundred year long dating coherency between the lower deep ice cores on Colle Gnifetti now comprises a firm basis for an investigation with respect to a common signal in the ice core records. This allows a fundamental assessment of the multi-core approach to paleoclimate reconstruction from Colle Gnifetti ice cores.

The success in extending the dating coherency is still restricted to a maximum of two centuries and accompanied by a relatively high uncertainty. An attempt to reach a consistent dating scenario for the cores at higher ages demands an improvement of at least one of the method's compartiments, i.e. ice core datings, GPR data or the model:

- A more sophisticated model: Especially the items listed in section 6.4 could be considered, namely accounting for firn compaction, three-dimensionality and input parameters beyond steady state.

- IRHs at greater depth: A possible identification of deeper reflections within the glacier body would allow to validate the method at higher ages.
- Revised ice core datings: The dating of the lower parts of the ice cores from Colle Gnifetti is subject to ongoing research [May, 2009]. Results from these studies might also influence the quality of the presented method by providing important boundary conditions or fixed points.

Combination of GPR, simple flow modeling and ice cores beyond Colle Gnifetti

The presented method should be easily applicable to other Alpine study sites for the derivation of a three-dimensional age distribution. The main requirements for a successful application of the method are

1. GPR profiles forming a closed polygon: They firstly define the area of interest, secondly provide information about accumulation rate and bedrock topography and thirdly can be used for validating the results if there are profiles parallel to flow lines.
2. information about surface topography (to be gained from GPS when recording GPR profiles).
3. at least one ice core for fixing the ages of the model isochrones and providing a vertical density distribution.
4. a flow model, e.g. SYNDICATE, which is already available and can easily be applied to other study sites.
5. surface velocity measurements: Although the measurements on Colle Gnifetti are only partially usable in this context, this is not necessarily the case at other sites. The more samples are available, the more adequate results might be gained from SYNDICATE, especially by taking into account the variable divergence implementation mentioned in section 6.4.

On Colle Gnifetti the numbers both of GPR profiles and of ice cores are extraordinarily high. If the method is applied to a site with a less dense data set, the uncertainties are naturally expected to be greater. Nevertheless, it can be a first approach to the age distribution at that site.

Bibliography

J. Alean, W. Haeberli, and B. Schädler. Snow accumulation, firn temperature and solar radiation in the area of the Colle Gnifetti core drilling site (Monte Rosa, Swiss Alps): distribution patterns and interrelationships. *Zeitschrift für Gletscherkunde und Glazialgeologie*, 19(2):131–147, 1984.

M. Armbruster. *Stratigraphische Datierung hoch-alpiner Eisbohrkerne über die letzten 1000 Jahre*. Diploma thesis, IUP Heidelberg, 2000.

R. Barnes. *Variogram Tutorial*. Golden Software, Inc., 2003. URL http://www.goldensoftware.com/variogramTutorial.pdf. last checked: 2011/02/07.

V. V. Bogorodsky, C. R. Bentley, and P. E. Gudmandsen. *Radioglaciology*. Glaciology and quaternary geology. D. Reidel Pub. Co., 1985.

P. Bohleber. *Age distribution and $\delta^{18}O$ variability in a low accumulation Alpine ice core: perspective for paleoclimate studies*. Diploma thesis, IUP Heidelberg, 2008.

R. Böhlert. *Glaziologische Untersuchungen auf dem Colle Gnifetti und auf dem Mt. Blanc: Ermittlung der Eisdickenverteilung und interner Schichten mittels Georadar*. Diploma thesis, Geographisches Institut der Universität Zürich, 2005.

J.-P. Chilès and P. Delfiner. *Geostatistics. Modeling Spatial Uncertainty*. Wiley series in probability and statistics. John Wiley & Sons, Inc., 1999.

W. Demtröder. *Experimentalphysik 2: Elektrizität und Optik*. Springer, 4th edition, 2006.

R. Drews, O. Eisen, I. Weikusat, S. Kipfstuhl, A. Lambrecht, D. Steinhage, F. Wilhelms, and H. Miller. Layer disturbances and the radio-echo free zone in ice sheets. *The Cryosphere*, 3:195–203, 2009.

O. Eisen, U. Nixdorf, L. Keck, and D. Wagenbach. Alpine ice cores and ground penetrating radar: combined investigations for glaciological and climatic interpretations of a cold Alpine ice body. *Tellus*, 55B:1007–1017, 2003.

O. Eisen, I. Hamann, S. Kipfstuhl, D. Steinhage, and F. Wilhelms. Direct evidence for continuous radar reflector originating from changes in crystal-orientation fabric. *The Cryosphere*, 1(1):1–10, 2007.

S. Fujita, T. Matsuoka, T. Ishida, K. Matsuoka, and S. Mae. A summary of the complex dielectric permittivity of ice in the megahertz range and its applications for radar sounding of polar ice sheets. In T. Hondoh, editor, *Physics of Ice Core Records*, pages 185–212, 2000.

W. Haeberli and J. Alean. Temperature and accumulation of high altitude firn in the Alps. *Annals of Glaciology*, 6:161–163, 1985.

W. Haeberli, H. P. Wächter, W. Schmid, and C. Sidler. Erste Erfahrungen mit dem U.S. Geological-Survey-Monopuls-Radioecholot im Firn, Eis und Permafrost der Schweizer Alpen. *Zeitschrift für Gletscherkunde und Glazialgeologie*, 19(1):61–72, 1983.

W. Haeberli, W. Schmid, and D. Wagenbach. On the geometry, flow and age of firn and ice at the Colle Gnifetti core drilling site (Monte Rosa, Swiss Alps). *Zeitschrift für Gletscherkunde und Glazialgeologie*, 24/1:1–19, 1988.

L. Hempel and F. Thyssen. Deep Radio Echo Soundings in the Vicinity of GRIP and GISP2 Drill Sites, Greenland. *Polarforschung*, 62(1):11–16, 1992.

M. Hoelzle, G. Darms, M. P. Lüthi, and S. Suter. Evidence of accelerated englacial warming in the Monte Rosa area, Switzerland/Italy. *The Cryosphere*, 5(1):231–243, 2011.

B. Hofmann-Wellenhof, H. Lichtenegger, and E. Wasle. *GNSS – Global Navigation Satellite Systems. GPS, GLONASS, Galileo and more.* SpringerWienNewYork, 2008.

J. D. Jackson. *Classical electrodynamics.* Wiley, 1962.

K. Jepsen. *Leitfähigkeits- und Dichtemessungen an einem alpinen Eiskern.* Bachelor thesis, Alfred-Wegener-Institut für Polar- und Meeresforschung, 2010.

L. Keck. *Climate significance of stable isotope records from Alpine ice cores.* PhD thesis, IUP Heidelberg, 2001.

P. K. Kitanidis. *Introduction to Geostatistics. Applications in hydrogeology.* Cambridge University Press, 1997.

A. Kovacs, A. J. Gow, and R. M. Morey. The in-situ dielectric constant of polar firn revisited. *Cold Regions Science and Technology*, 23:245–256, 1995.

F. Lambert, B. Delmonte, J. R. Petit, M. Bigler, P. R. Kaufmann, M. A. Hutterli, T. F. Stocker, U. Ruth, J. P. Steffensen, and V. Maggi. Dust-climate couplings over the past 800,000 years from the EPICA Dome C ice core. *Nature*, 452: 616–619, 2008.

D. Lüthi, M. Le Floc, B. Bereiter, T. Blunier, J.-M. Barnola, U. Siegenthaler, D. Raynaud, J. Jouzel, H. Fischer, K. Kawamura, and T. F. Stocker. High-resolution carbon dioxide concentration record 650,000–800,000 years before present. *Nature*, 453:379–382, 2008.

M. P. Lüthi. *Rheology of cold firn and dynamics of a polythermal ice stream.* PhD thesis, Versuchsanstalt für Wasserbau, Hydrologie und Glaziologie der ETH Zürich, 2000.

M. P. Lüthi and M. Funk. Dating ice cores from a high Alpine glacier with a flow model for cold firn. *Annals of Glaciology*, 31:69–79, 2000.

M. P. Lüthi and M. Funk. Modelling heat flow in a cold, high altitude glacier: interpretation of measurements from Colle Gnifetti, Swiss Alps. *Journal of Glaciology*, 47(157):314–324, 2001.

B. May. *Radiocarbon microanalysis on ice impurities for dating of Alpine glaciers.* PhD thesis, IUP Heidelberg, 2009.

M. L. Moran, R. J. Greenfield, S. A. Arcone, and A. J. Delaney. Delineation of a complexly dipping temperate glacier bed using short-pulse radar arrays. *Journal of Glaciology*, 46(153):274–286, 2000.

F. Navarro and O. Eisen. Ground-penetrating radar in glaciological applications. In P. Pellika and G. Rees, editors, *Remote Sensing of Glaciers*, chapter 11. Taylor and Francis, 2009.

H. Fischer, M. Werner, D. Wagenbach, M. Schwager, T. Thorsteinnson, F. Wilhelms, J. Kipfstuhl, and S. Sommer. Little ice age clearly recorded in northern Greenland ice cores. *Geophysical Research Letters*, 25(10):1749–1752, 1998.

H. Oeschger, U. Schotterer, W. Haeberli, and H. Röthlisberger. First results from Alpine core drilling projects. *Zeitschrift für Gletscherkunde und Glazialgeologie*, 13(1/2):193–208, 1977.

W. S. B. Paterson. *The Physics of Glaciers.* Pergamon, 3rd edition, 1994.

V. F. Petrenko and R. W. Whitworth. *Physics of Ice.* Oxford University Press, 1999.

J. Schäfer. *Rekonstruktion bio-chemischer Spurenstoffkreisläufe anhand eines alpinen Eisbohrkerns.* Diploma thesis, IUP Heidelberg, 1995.

U. Siegenthaler, T. F. Stocker, E. Monnin, D. Lüthi, J. Schwander, B. Stauffer, D. Raynaud, J.-M. Barnola, H. Fischer, V. Masson-Delmotte, and J. Jouzel. Stable Carbon Cycle-Climate Relationship During the Late Pleistocene. *Science*, 310(5752):1313–1317, 2005.

C. Vincent, M. Vallon, J. F. Pinglot, M. Funk, and L. Reynaud. Snow accumulation and ice flow at Dôme du Goûter (4300m), Mont Blanc, French Alps. *Journal of Glaciology*, 43(145):513–521, 1997.

H. Wackernagel. *Multivariate Geostatistics.* Springer, 1995.

D. Wagenbach. Environmental Records in Alpine Glaciers. In H. Oeschger and C. C. Langway, Jr., editors, *The Environmental Record in Glaciers and Ice Sheets*, pages 69–83. John Wiley and Sons, 1989.

D. Wagenbach. Special problems of mid-latitude glacier ice-core research. In W. Haeberli and B. Stauffer, editors, *Greenhouse Gases, Isotopes and Trace Elements in Glaciers as Climatic Evidence of the Holocene*, volume Arbeitsheft Nr. 14 of *Report of the ESF/EPC Workshop, Zürich 27-28 October 1992.* Versuchsanstalt für Wasserbau, Hydrologie und Glaziologie der Eidgenössischen Technischen Hochschule Zürich, 1992.

D. Wagenbach, editor. *Environmental and Climatic Records from High Elevation Alpine Glaciers (ALPCLIM).* Final Report, 2001.

S. Wagner. *Dreidimensionale Modellierung zweier Gletscher und Deformationsanalyse von eisreichem Permafrost.* PhD thesis, Versuchsanstalt für Wasserbau, Hydrologie und Glaziologie der ETH Zürich, 1996.

F. Wilhelms. Leitfähigkeits- und Dichtemessung an Eisbohrkernen. *Berichte zur Polarforschung*, 191, 1996.

Ö. Yilmaz. *Seismic data analysis: processing, inversion, and interpretation of seismic data.* Number 10 in Investigations in geophysics. Society of Exploration Geophysicists, 2nd edition, 2001.

Lists

List of Figures

List of Tables

List of acronyms and designations

CC	ice core on Colle Gnifetti
CMP	common midpoint
CO	common offset
COURAN	flow model as presented by Vincent et al. (1997)
ECK	intersection of radar profiles T1 and F4
EM	electromagnetic
GPR	ground-penetrating radar
GPS	global positioning system
IRH	internal reflection horizon
IS	intersection
KCH	ice core on Colle Gnifetti
KCI	ice core on Colle Gnifetti
KCS	ice core on Colle Gnifetti
STD	standard deviation
SYNDICATE	model developed for this thesis basing on COURAN
TWT	two-way travel time
we	water equivalent

Appendices

A Appendix to chapter 3

A.1 COURAN and SYNDICATE: exemplary input and output

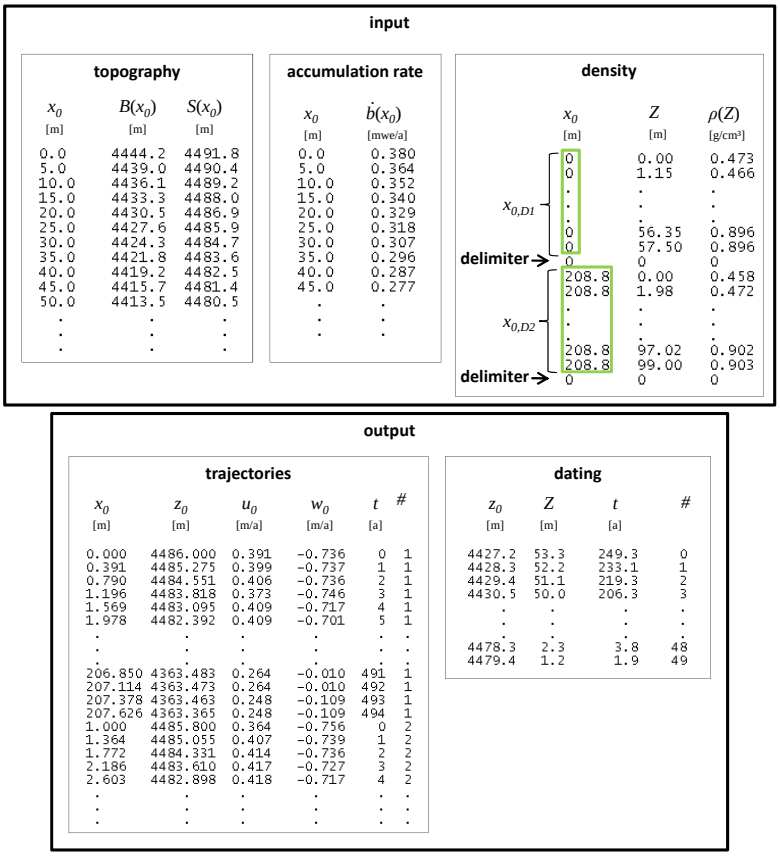


Figure A.1
Exemplary input and output; # means number of trajectory within that run.

A.2 Ice slab vs. SYNDICATE

Refering to the approximations explained in section 3.4 (Figure 3.8), a direct comparison between the ice slab model and SYNDICATE is carried out: For this

purpose SYNDICATE is applied to an ice slab characterized by

- constant density $\rho = 0.9\,\frac{\mathrm{g}}{\mathrm{cm}^3}$
- constant accumulation rate $\dot{b} = 1\,\frac{\mathrm{mwe}}{\mathrm{a}}$
- constant surface and bed slope α and β.

For four combinations of α and β, the results are shown in Figure A.2. As it is expected, there is no difference if $\alpha = \beta = 0$ (a). The effect of the approximations on the trajectories increases with increasing α. This is due to the evaluation of q at x_0 instead of x in SYNDICATE[12]: The corresponding q is lower because the distance from $x_0 = 0$ to the point vertically above the considered point in the glacier, along which mass is accumulated, is lower than that from $x_0 = 0$ to the surface point perpendicular above the considered point. The isochrones are rather unaffected which is due to constant accumulations rate.

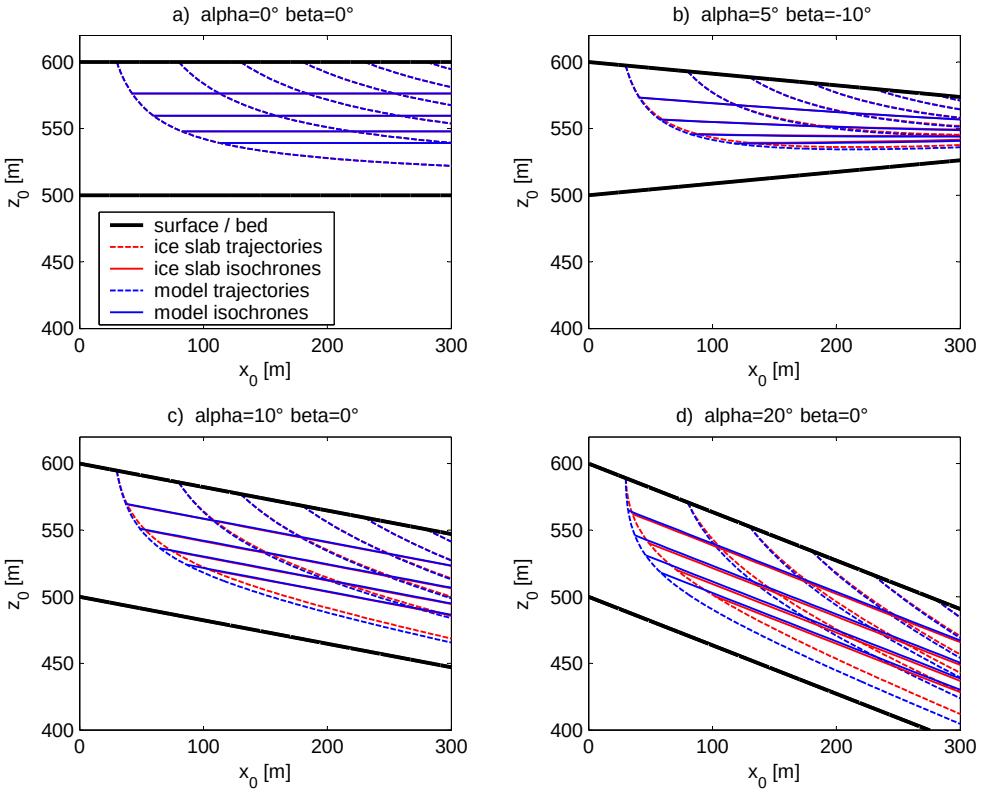


Figure A.2
Ice slab vs. SYNDICATE for four surface and bed angles α, β.

[12]The evaluation of $\dot{b}$ at x_0 has no effect since $\dot{b} = \text{const}$.

A.3 SYNDICATE: Sensitivity

SYNDICATE is applied to an exemplary and fictive flow line for studying its sensitivity to the input parameters. The input data are shown in Figure A.3 and Table A.1.

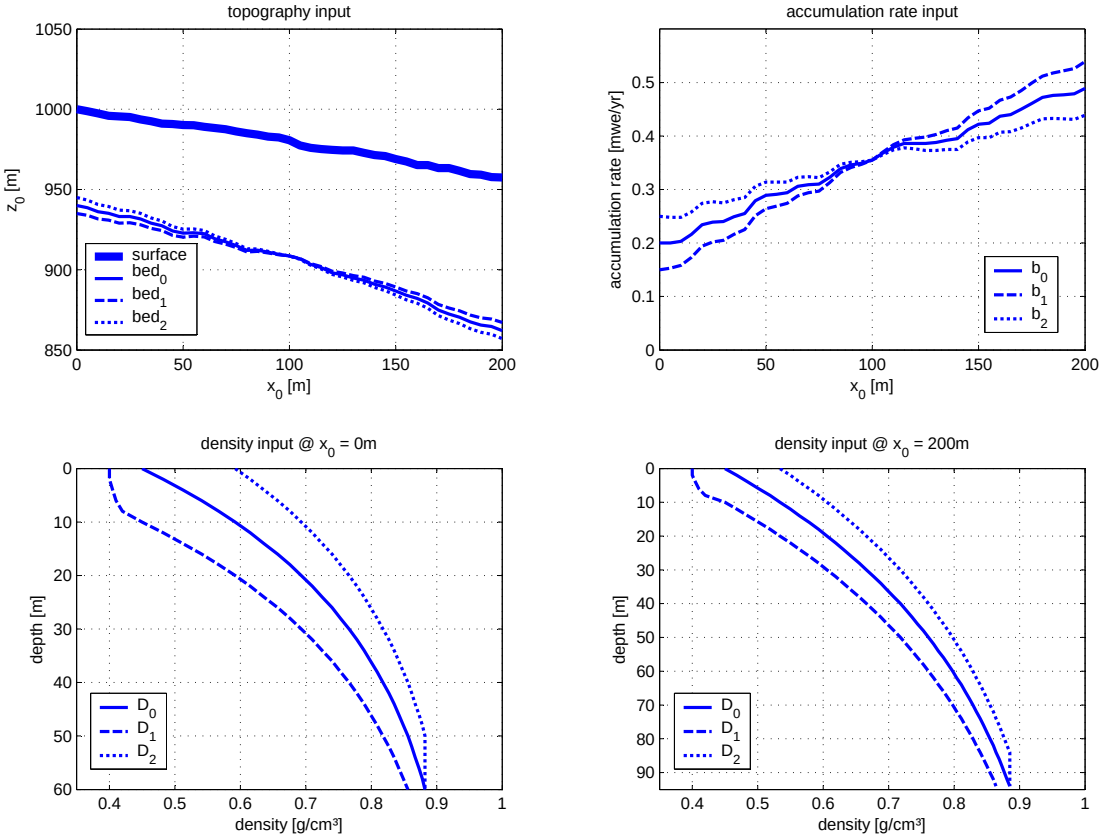


Figure A.3
Sensitivity: input data 1; index 0 indicates the 'original' input data that are each varied twice (index 1 and 2).

- The variation of glacier bed is obtained by adding ΔB linear in x_0 that results in ± 5 m at $x_0 = 0$ m and $x_0 = 200$ m respectively.
- $\dot{b}$ is varied in the same way: $\Delta\dot{b} = \pm 0.05 \frac{\text{mwe}}{\text{a}}$ at $x_0 = 0$ m, 200 m and linear between these points.
- The varied density distribution are obtained by shifting the original distribution by ± 10 m.

D_0	D_1	D_2	$q_{0,0}$	$q_{0,1}$	$q_{0,2}$
0.4	0.3	0.5	$15\frac{\text{m mwe}}{\text{a}}$	$10\frac{\text{m mwe}}{\text{a}}$	$20\frac{\text{m mwe}}{\text{a}}$

Table A.1
Sensitivity: input data 2; index 0 indicated the 'original' input data that are each varied twice (index 1 and 2).

In the following, the results are illustrated (except for the sensitivity to accumulation rate: chapter 3). Each plot shows the model's sensitivity to one input parameter. The other parameters are those with index 0. The illustrated trajectories are those starting at $x_0 = 0$, 50, 100, 150 m. The isochrones are those of age $t = 10$, 50, 100, 200 a. The illustration is analogous to that in Figure 3.12: The indices in output illustration refer to the indices of input data (e.g. in Figure A.4: the dashed isochrones and trajectories (index 1) result from the model run with the bedrock altitude set 1 in Figure A.3 (left top – dashed line as well). This implies that the output data indicated by 0 are the same in each plot because the underlying input data are the same.

Glacier bed topography

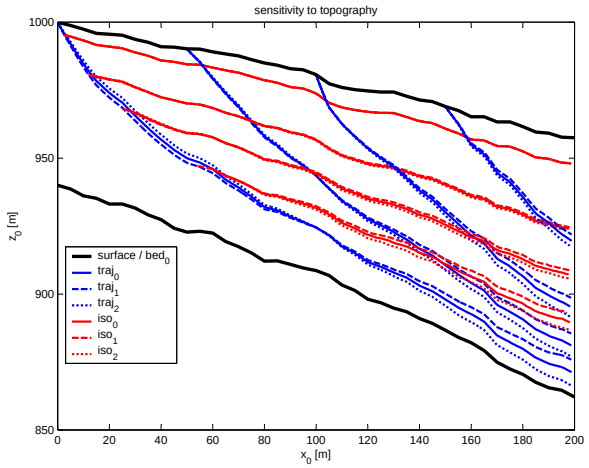


Figure A.4
Sensitivity to glacier bed variation.

The bed topography affects both isochrones and trajectories. The effect is greater near the bed, as it is expected: u_0 and w_0 are polynomials in $\frac{Z_{we}}{H_{we}}$.

Density

Density variation affects the trajectories only little because both components u_0 and w_0 are scaled by it. The conversion of Z and H is the only reason for variation in the upper region. Near the bedrock the kinematic correction results in higher

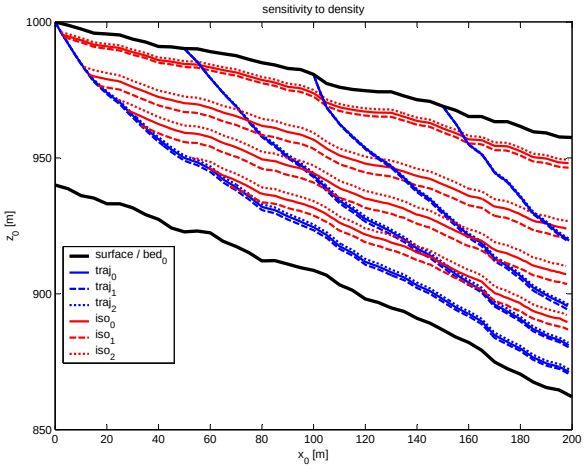


Figure A.5
Sensitivity to density variation.

discrepancies. The isochrones are affected much more because their depths are governed by w which surely depends on the scaling by density.

Divergence

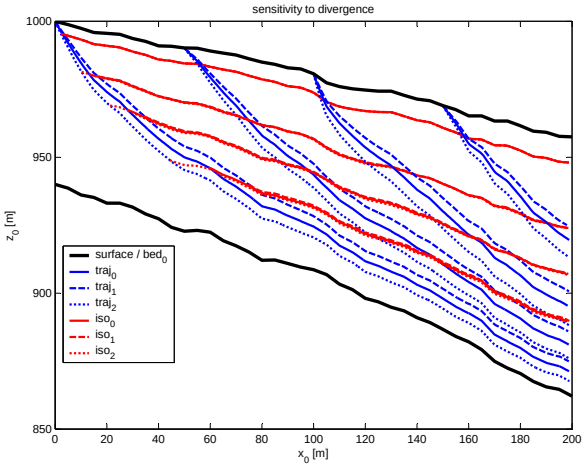


Figure A.6
Sensitivity to divergence variation.

The divergence parameter D shifts the trajectories because the surface parallel velocity u is proportional to $q \propto (1 - D)$. Thus, high divergence results in trajectories with lower surface parallel component. The isochrones do not depend on q (only via the rotation of coordinates and close to bedrock via the kinematic correction) and are thus not affected that much by variations in D.

Flux from uphill direction

The effects of q_0-variation and the explanations of these are the same as in the case of divergence. The only difference is that trajectories starting at large distances from $x_0 = 0$, the q_0-variations has less effect than the D-variation because the former is a unique shift of q, whose relative effect decreases with x_0, while the latter lowers q at each x_0.

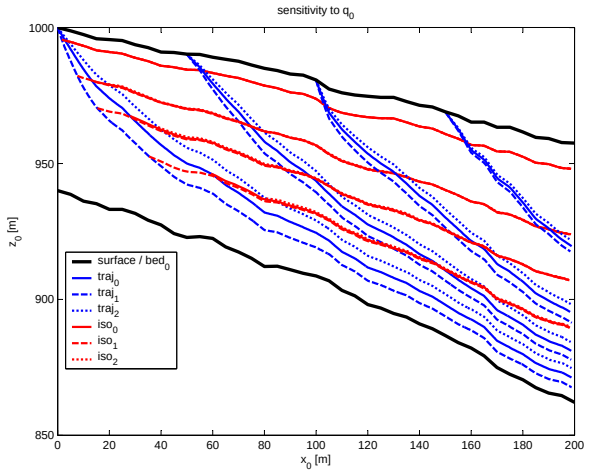


Figure A.7
Sensitivity to q_0 variation.

Dating variation (all input parameters)

The range of ages at $x_{0,c} = 190$ m at each depth according to the input variations explained above: Since the age-depth relation is directly related to the isochrones,

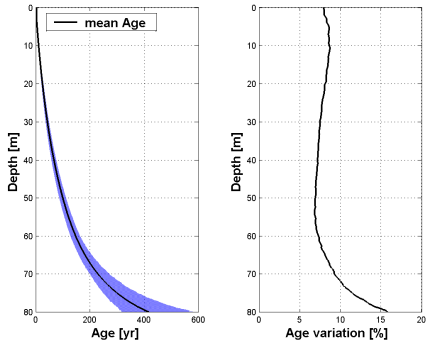


Figure A.8
Sensitivity of modeled ice core dating.

the sensitivity of the age-depth relation at the model drilling site is governed by the same parameters as the isochrones making D and q_0 rather indecisive.

A.4 Bootstrapping

The bootstrapping routine for estimating the error propagation in SYNDICATE is performed with Matlab. It runs SYNDICATE a defined number of times (usually 500 times) with varied input. In the following, the input variation and the output analysis is presented.

Input variation

- **Ice thickness:** Two types of variation are applied to the bedrock topography
  1. Offset: The mean error ΔH_m of the ice thickness is calculated along one flow line. Then one global offset for H at all x_0-values is generated randomly. The underlying distribution is normal with zero mean and standard deviation (STD) $\Delta H_\mathrm{m}/2$. This leads to a global increase or decrease of ice thickness all over the flow line.
  2. Varying bed slope: The ice thickness at the two edges $x_{0,1} = 0$ and $x_{0,2} = x_{0,\mathrm{max}}$ of the flow line is varied randomly by a normal distribution with STDs $\Delta H(x_{0,1})/2$ and $\Delta H(x_{0,2})/2$ respectively and zero mean. Then this variations is interpolated linearly in x_0 and added to the ice thickness in order to obtain varied bed slopes.

- **Surface altitute:** In order to account for possible higher or lower surface slopes, the surface altitude is varied corresponding to the second type of ice thickness variation. Here, the two STDs are $\Delta S(x_{0,1}) = \Delta S(x_{0,2}) = 1$ m. A further global offset was rejected because it is already taken into account by the ice thickness variation.

- **Accumulation rate:** The accumulation rate is varied corresponding to the ice thickness.

- **Density:** At both density input locations ($x_{0,D1}$ and $x_{0,D2}$), Δz_1 and Δz_2 respectively[13] are added to the depth coordinates of the according density distribution. If $\Delta z_i < 0$, the part above $z = 0$ is deleted and the missing values in the lower part are extrapolated linearly from the undermost two values up to a maximum of 0.9 g/cm^3 (or with ρ_max of the density input, if $\rho_\mathrm{max} > 0.9$ g/cm^3). If $\Delta z_i > 0$, the part below $z = z_\mathrm{max}$ (of the original density input data) is deleted. The data are extrapolated towards $z = 0$ linearly down to a mininum of 0.35 g/cm^3.

[13] generated from a normal distribution with zero mean and STD of 5 m

- **Flux parameters:** q_0 and D are both varied normally with means and STDs as discussed in section 5.3. The range of D is restricted to the interval [0, 0.8] because $D < 0$ means convergence of flow lines which is not given on Colle Gnifetti and $D > 0.8$ makes the code unstable due to small surface parallel velocity components. q_0 is restricted to values greater than zero.

Output averaging

- The altitudes of isochrones and trajectories from each model run are interpolated on a x_0-grid. Then at each value of x_0 the altitude z_0 of each isochrone / trajectory is averaged and the standard deviation of the z_0-values is calculated as output error. The altitude values were sampled for checking wether they are distributed normally which would justify this consideration of output error. Some of these samples are shown in Figures A.9 and A.10.

- The modeled ice core datings are interpolated on a depth grid and then averaged at each depth value.

- The re-arranging of trajectories for source region output is done by

  1. re-ordering the trajectories with respect to depth intervals at the model drilling site $x_{0,c}$
  2. for each depth interval: averaging of x_0-coordinate of the trajectories' starting points ($\Rightarrow$ mean $\overline{x_{0,\mathrm{s}}}$ and standard deviation $\Delta\overline{x_{0,\mathrm{s}}}$)[14]
  3. collecting all trajectories starting near $x_{0,1} = \overline{x_{0,\mathrm{s}}} - \Delta\overline{x_{0,\mathrm{s}}}$, $x_{0,2} = \overline{x_{0,\mathrm{s}}}$ and $x_{0,3} = \overline{x_{0,\mathrm{s}}} + \Delta\overline{x_{0,\mathrm{s}}}$ (and ending in the specific depth interval of course)
  4. for $i = 1, 2, 3$: averaging each of these three sets of collected trajectories.

  For each depth interval the three averaged trajectories are assumed to represent the mean path (index 2) and the confidence intervals (indices 1 and 3).

[14] Index s means trajectory starting point.

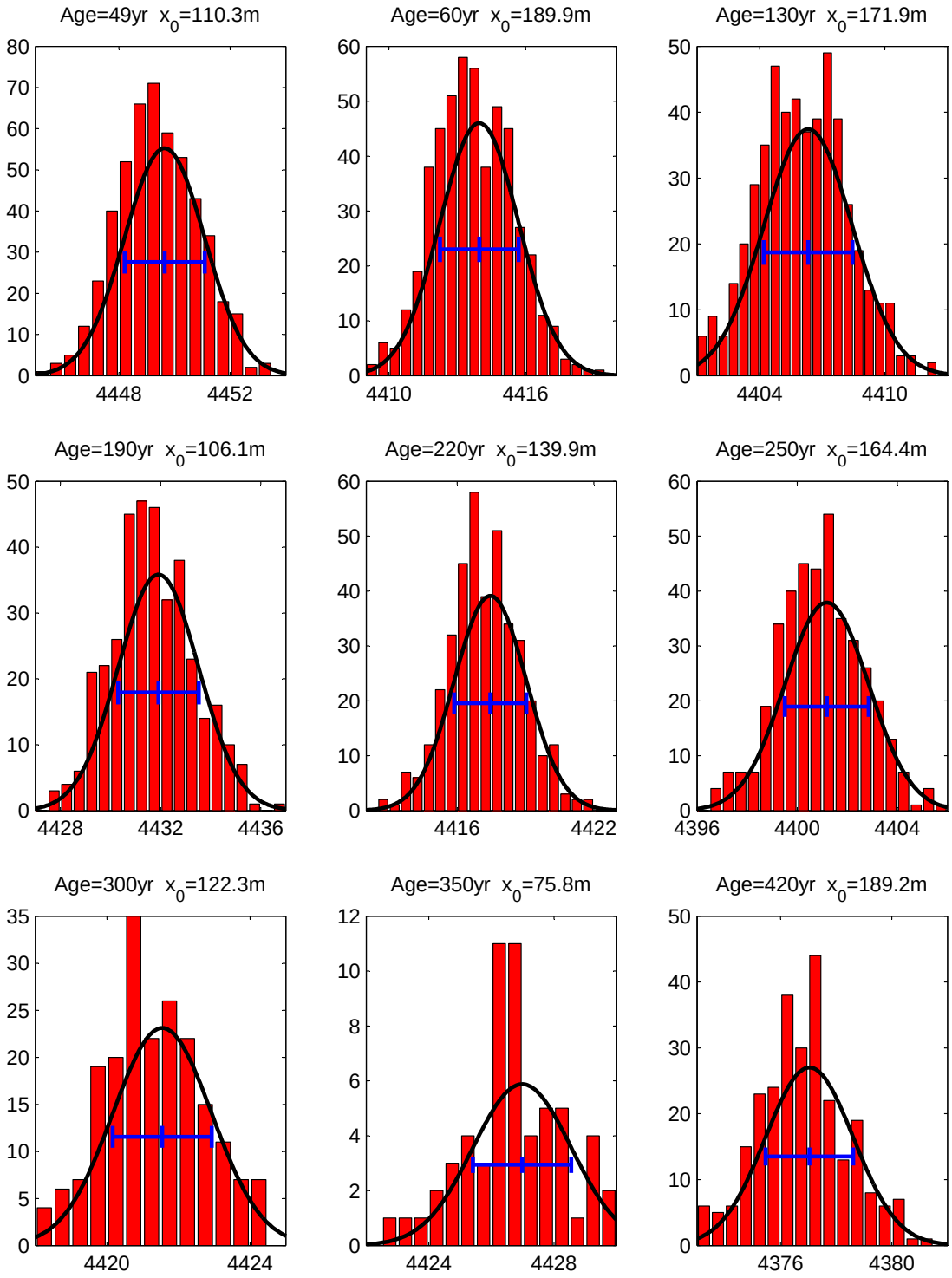


Figure A.9
Histograms of isochrone altitudes (isochrone and x_0 sampled randomly) on the flow line from KCH to KCS; the Gaussian distributions are not fitted to the data but are those corresponding to the mean and the variance of the data sets.

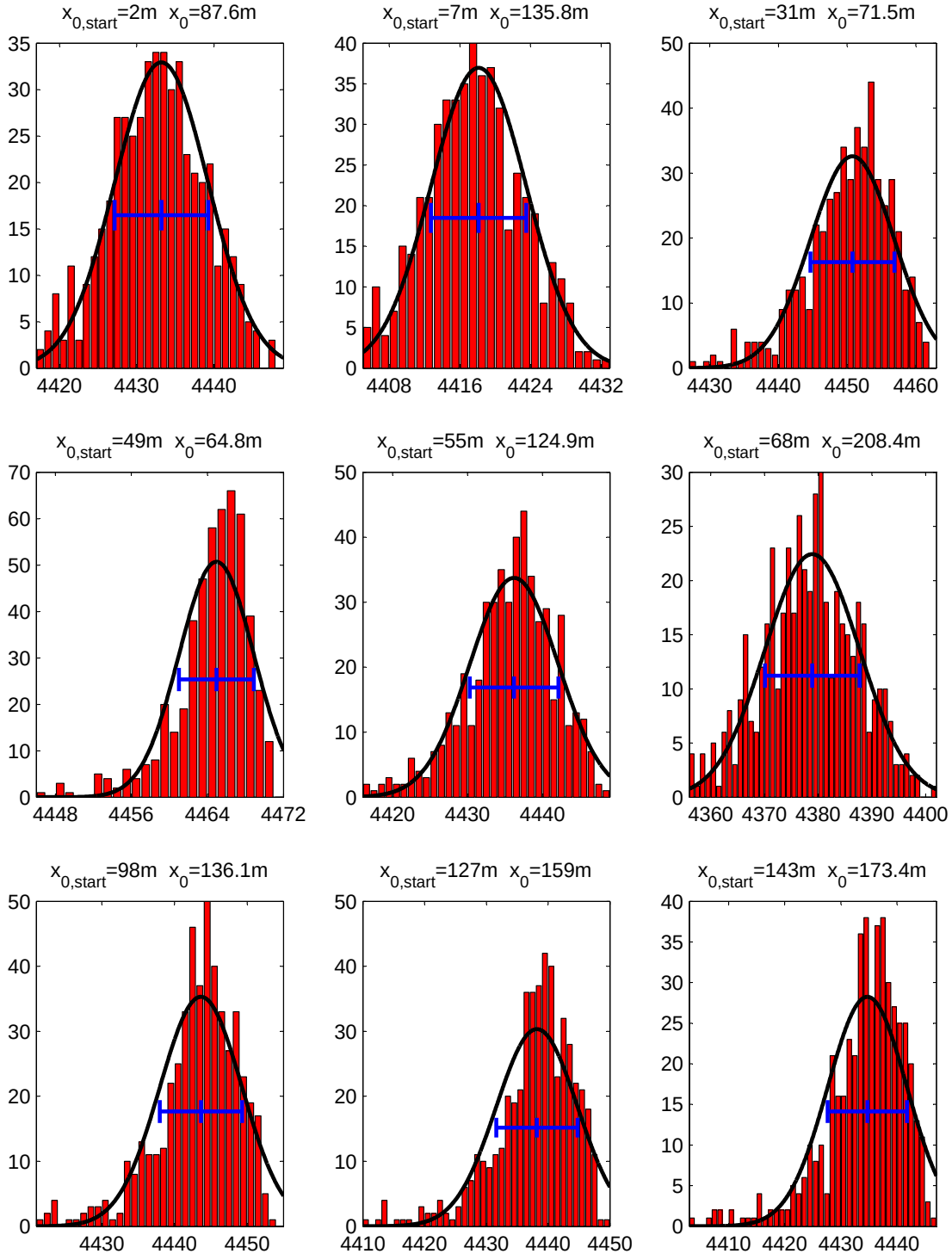


Figure A.10
Histograms of trajectory altitudes (trajectory and x_0 sampled randomly) on the flow line from KCH to KCS; the Gaussian distributions are not fitted to the data but are those corresponding to the mean and the variance of the data sets. The Gaussian distributions fit worse to the data than in the case of isochrones.

B Appendix to chapter 4

B.1 TWTs at the intersections

The compilation of TWTs at the intersections provides an insight to the IRH coherency on the different GPR profiles.
Since F4 and T1 were recorded in sequence, they are treated as one profile refered to as P41 in the following. The abbreviations in the tables are

IS: intersection / ice core
P1: profile 1 of the IS
P2: profile 2 of the IS
tr1: trace of profile 1 at the IS
tr2: trace of profile 2 at the IS
t1: t_{TWT} on profile 1 [ns]
t2: t_{TWT} on profile 2 [ns]

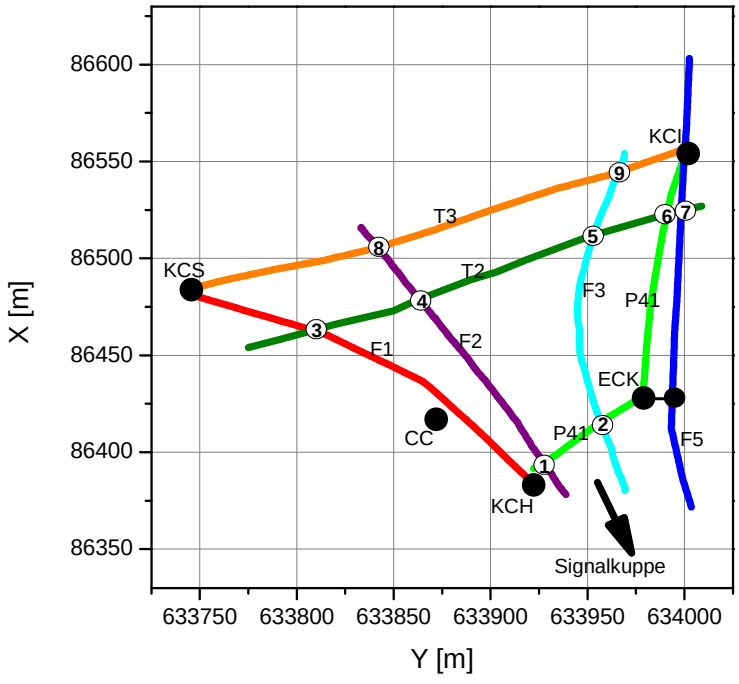


Figure B.1
Intersections of GPR profiles; Note P41 instead of F4 and T1.

	IS	P1	tr1	t1	P2	tr2	t2	t1-t2
1	KCH	F1	418	78	P41	408	77	1
	KCS	F1	6	127	T3	551	128	-1
	3	F1	140	119	T2	132	121	-2
	KCI	P41	5	56	T3	1	40	16
	2	P41	322	71	F3	279	68	3
	1	P41	396	77	F2	283	76	1
	6	P41	77	50	T2	12	44	6
	9	T3	79	52	F3	20	51	1
	8	T3	348	97	F2	26	97	0
	5	F3	88	56	T2	37	62	-6
	4	F2	88	79	T2	97	96	-17
	KCI	F5	280	47	P41	16	55	-8
	ECK	F5	950	68	P41	271	69	-1
	KCI	F5	255	48	T3	4	41	7

	IS	P1	tr1	t1	P2	tr2	t2	t1-t2
2	KCH	F1	418	93	P41	408	91	2
	KCS	F1	6	166	T3	551	165	1
	3	F1	140	150	T2	132	155	-5
	KCI	P41	5	60	T3	1	61	-1
	2	P41	322	87	F3	279	86	1
	1	P41	396	91	F2	283	101	-10
	6	P41	77	57	T2	12	62	-5
	9	T3	79	77	F3	20	65	12
	8	T3	348	124	F2	26	123	1
	5	F3	88	66	T2	37	88	-22
	4	F2	88	102	T2	97	126	-24
	KCI	F5	280	51	P41	16	60	-9
	ECK	F5	950	84	P41	271	84	0
	KCI	F5	255	52	T3	4	62	-10

	IS	P1	tr1	t1	P2	tr2	t2	t1-t2
3	KCH	F1	418	99	P41	408	98	1
	KCS	F1	6	185	T3	551	184	1
	3	F1	140	164	T2	132	170	-6
	KCI	P41	5	66	T3	1	66	0
	2	P41	322	92	F3	279	93	-1
	1	P41	396	97	F2	283	105	-8
	6	P41	77	68	T2	12	67	1
	9	T3	79	80	F3	20	75	5
	8	T3	348	133	F2	26	133	0
	5	F3	88	76	T2	37	94	-18
	4	F2	88	113	T2	97	138	-25
	KCI	F5	280	61	P41	16	66	-5
	ECK	F5	950	92	P41	271	91	1
	KCI	F5	255	62	T3	4	68	-6

	IS	P1	tr1	t1	P2	tr2	t2	t1-t2
4	KCH	F1	418	116	P41	408	115	1
	KCS	F1	6	217	T3	551	214	3
	3	F1	140	195	T2	132	199	-4
	KCI	P41	5	72	T3	1	78	-6
	2	P41	322	112	F3	279	111	1
	1	P41	396	115	F2	283	120	-5
	6	P41	77	72	T2	12	92	-20
	9	T3	79	90	F3	20	89	1
	8	T3	348	147	F2	26	147	0
	5	F3	88	92	T2	37	124	-32
	4	F2	88	125	T2	97	171	-46
	KCI	F5	280	73	P41	16	73	0
	ECK	F5	950	109	P41	271	109	0
	KCI	F5	255	75	T3	4	79	-4

	IS	P1	tr1	t1	P2	tr2	t2	t1-t2
5	KCH	F1	418	135	P41	408	135	0
	KCS	F1	6	253	T3	551	255	-2
	3	F1	140	226	T2	132	225	1
	KCI	P41	5	77	T3	1	83	-6
	2	P41	322	133	F3	279	131	2
	1	P41	396	136	F2	283	144	-8
	6	P41	77	81	T2	12	112	-31
	9	T3	79	105	F3	20	99	6
	8	T3	348	173	F2	26	171	2
	5	F3	88	108	T2	37	143	-35
	4	F2	88	145	T2	97	197	-52
	KCI	F5	280	78	P41	16	77	1
	ECK	F5	950	129	P41	271	130	-1
	KCI	F5	255	82	T3	4	85	-3

	IS	P1	tr1	t1	P2	tr2	t2	t1-t2
6	KCH	F1	418	158	P41	408	159	-1
	KCS	F1	6	303	T3	551	302	1
	3	F1	140	265	T2	132	265	0
	KCI	P41	5	87	T3	1	99	-12
	2	P41	322	153	F3	279	151	2
	1	P41	396	158	F2	283	172	-14
	6	P41	77	92	T2	12	131	-39
	9	T3	79	125	F3	20	109	16
	8	T3	348	202	F2	26	201	1
	5	F3	88	117	T2	37	173	-56
	4	F2	88	171	T2	97	233	-62
	KCI	F5	280	86	P41	16	87	-1
	ECK	F5	950	147	P41	271	147	0
	KCI	F5	255	89	T3	4	103	-14

	IS	P1	tr1	t1	P2	tr2	t2	t1-t2
7	KCH	F1	418	183	P41	408	186	-3
	KCS	F1	6	351	T3	551	352	-1
	3	F1	140	299	T2	132	300	-1
	KCI	P41	5	102	T3	1	109	-7
	2	P41	322	184	F3	279	188	-4
	1	P41	396	186	F2	283	199	-13
	6	P41	77	113	T2	12	160	-47
	9	T3	79	139	F3	20	138	1
	8	T3	348	231	F2	26	231	0
	5	F3	88	153	T2	37	208	-55
	4	F2	88	194	T2	97	266	-72
	KCI	F5	280	99	P41	16	101	-2
	ECK	F5	950	180	P41	271	181	-1
	KCI	F5	255	103	T3	4	111	-8

	IS	P1	tr1	t1	P2	tr2	t2	t1-t2
8	KCH	F1	418	202	P41	408	200	2
	KCS	F1	6	393	T3	551	393	0
	3	F1	140	327	T2	132	-	-
	KCI	P41	5	112	T3	1	114	-2
	2	P41	322	200	F3	279	203	-3
	1	P41	396	201	F2	283	224	-23
	6	P41	77	127	T2	12	-	-
	9	T3	79	149	F3	20	148	1
	8	T3	348	253	F2	26	251	2
	5	F3	88	162	T2	37	-	-
	4	F2	88	213	T2	97	-	-
	KCI	F5	280	108	P41	16	113	-5
	ECK	F5	950	199	P41	271	199	0
	KCI	F5	255	111	T3	4	116	-5

	IS	P1	tr1	t1	P2	tr2	t2	t1-t2
9	KCH	F1	418	227	P41	408	224	3
	KCS	F1	6	434	T3	551	436	-2
	3	F1	140	352	T2	132	-	-
	KCI	P41	5	113	T3	1	129	-16
	2	P41	322	224	F3	279	222	2
	1	P41	396	224	F2	283	258	-34
	6	P41	77	133	T2	12	-	-
	9	T3	79	178	F3	20	148	30
	8	T3	348	281	F2	26	281	0
	5	F3	88	170	T2	37	-	-
	4	F2	88	240	T2	97	-	-
	KCI	F5	280	119	P41	16	115	4
	ECK	F5	950	217	P41	271	217	0
	KCI	F5	255	121	T3	4	132	-11

	IS	P1	tr1	t1	P2	tr2	t2	t1-t2
10	KCH	F1	418	244	P41	408	246	-2
	KCS	F1	6	458	T3	551	456	2
	3	F1	140	371	T2	132	-	-
	KCI	P41	5	122	T3	1	139	-17
	2	P41	322	250	F3	279	249	1
	1	P41	396	247	F2	283	274	-27
	6	P41	77	146	T2	12	-	-
	9	T3	79	190	F3	20	167	23
	8	T3	348	295	F2	26	294	1
	5	F3	88	192	T2	37	-	-
	4	F2	88	253	T2	97	-	-
	KCI	F5	280	119	P41	16	125	-6
	ECK	F5	950	246	P41	271	245	1
	KCI	F5	255	121	T3	4	141	-20

	IS	P1	tr1	t1	P2	tr2	t2	t1-t2
11	KCH	F1	418	278	P41	408	276	2
	KCS	F1	6	497	T3	551	496	1
	3	F1	140	396	T2	132	-	-
	KCI	P41	5	137	T3	1	163	-26
	2	P41	322	279	F3	279	279	0
	1	P41	396	277	F2	283	318	-41
	6	P41	77	163	T2	12	-	-
	9	T3	79	220	F3	20	186	34
	8	T3	348	326	F2	26	324	2
	5	F3	88	215	T2	37	-	-
	4	F2	88	284	T2	97	-	-
	KCI	F5	280	133	P41	16	140	-7
	ECK	F5	950	276	P41	271	278	-2
	KCI	F5	255	133	T3	4	167	-34

	IS	P1	tr1	t1	P2	tr2	t2	t1-t2
BR	KCH	F1	418	629	P41	408	632	-3
	KCS	F1	6	1102	T3	551	1102	0
	3	F1	140	717	T2	132	726	-9
	KCI	P41	5	645	T3	1	680	-35
	2	P41	322	711	F3	279	715	-4
	1	P41	396	648	F2	283	652	-4
	6	P41	77	691	T2	12	700	-9
	9	T3	79	786	F3	20	755	31
	8	T3	348	750	F2	26	727	23
	5	F3	88	736	T2	37	717	19
	4	F2	88	707	T2	97	703	4
	KCI	F5	280	648	P41	16	644	4
	ECK	F5	950	661	P41	271	747	-86
	KCI	F5	255	644	T3	4	676	-32

Table B.1
TWTs at the intersections of GPR profiles; each horizon is treated in one box. The last table (BR) is the bedrock reflection. Explanation of abbreviations: previous page; grey background: Trace tr2 of profile P2 is the starting point for tracking the phase. Thus, the differences must inevitably be low at these intersections. Red background: large TWT differences at the intersections; they mainly occur on T2 and on the deeper IRH near KCI.

B.2 Radargrams, IRHs and bedrock reflection

In the following the radargrams of the GPR profiles including the picked IRHs and the selected bedrock reflections are illustrated (except T3 – figure 4.10 on page 48). The radargrams are those which have been passed through the complete processing routine discussed in section 4.4.
Blue lines: IRHs; red solid line: bedrock reflection; red dashed lines: horizontal positions of intersections with other profiles. The intersection numbers are those in Figures 4.8 and B.1.

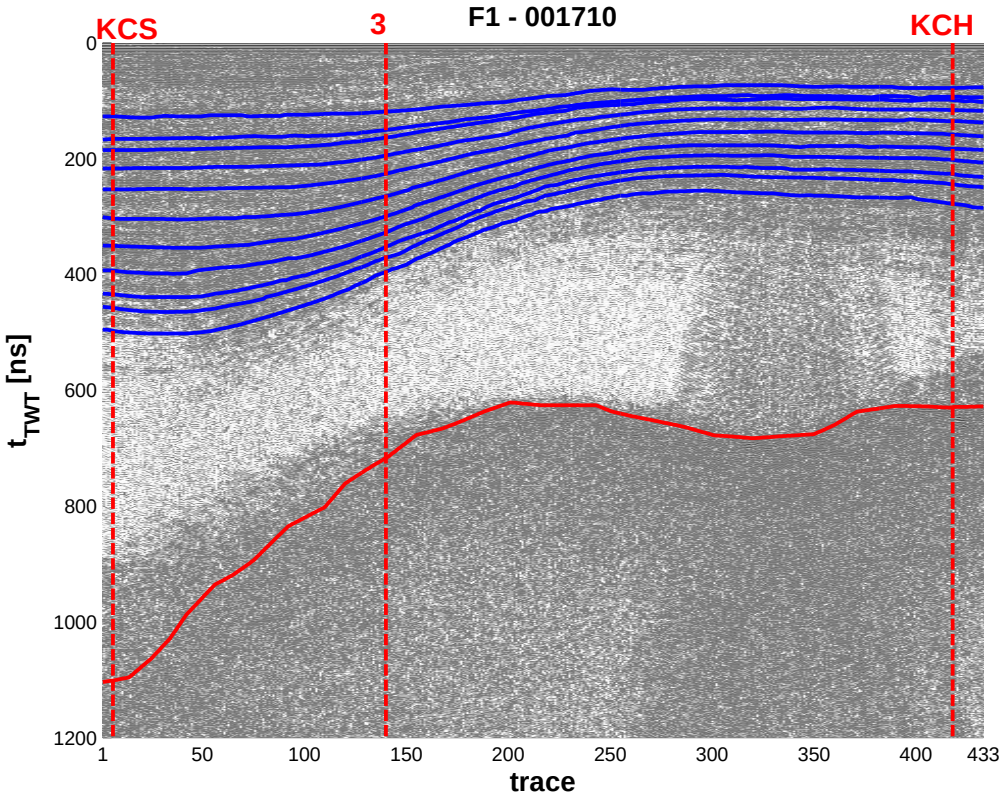


Figure B.2
Radargram of F1.

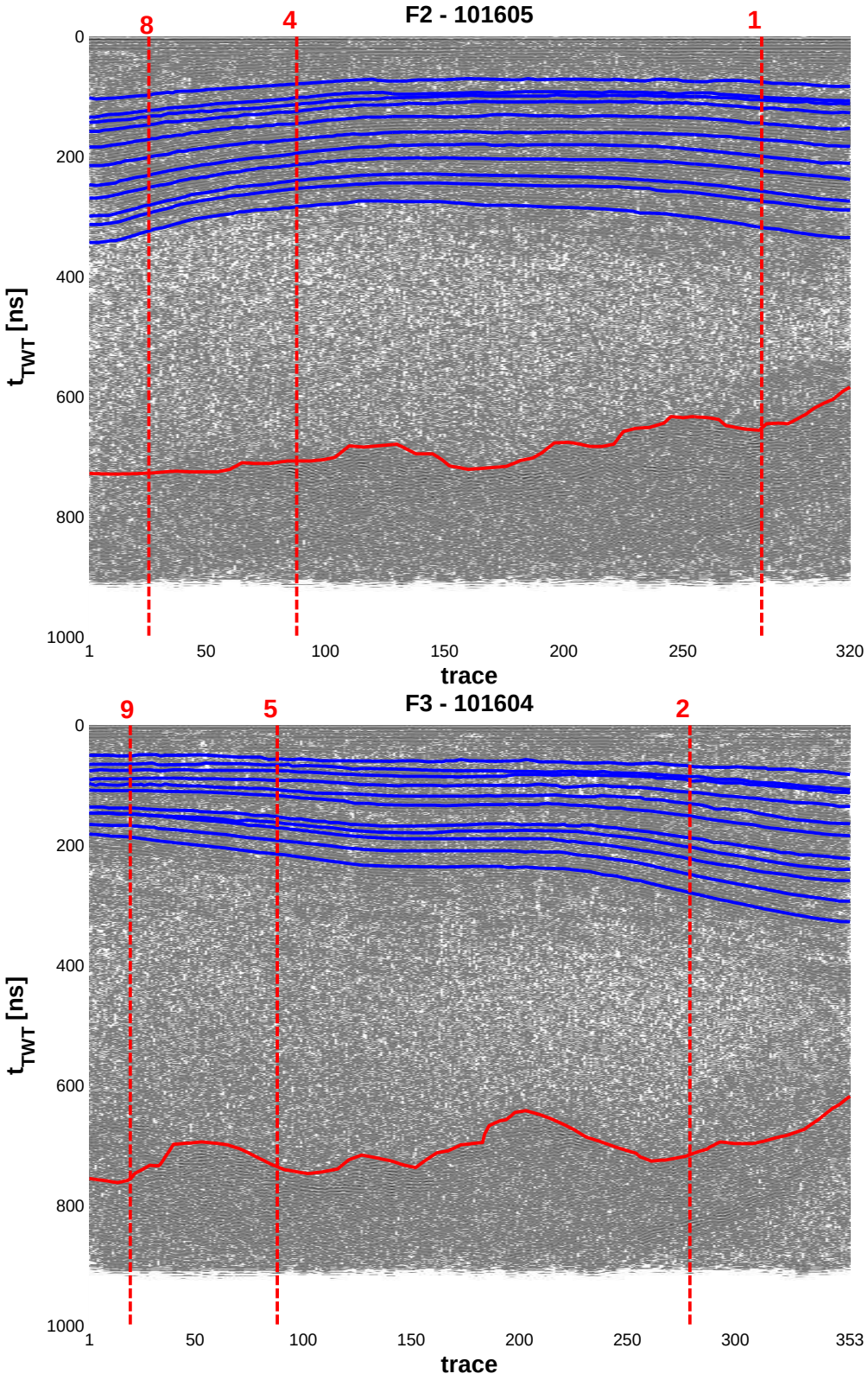


Figure B.3
Radargram of F2 and F3.

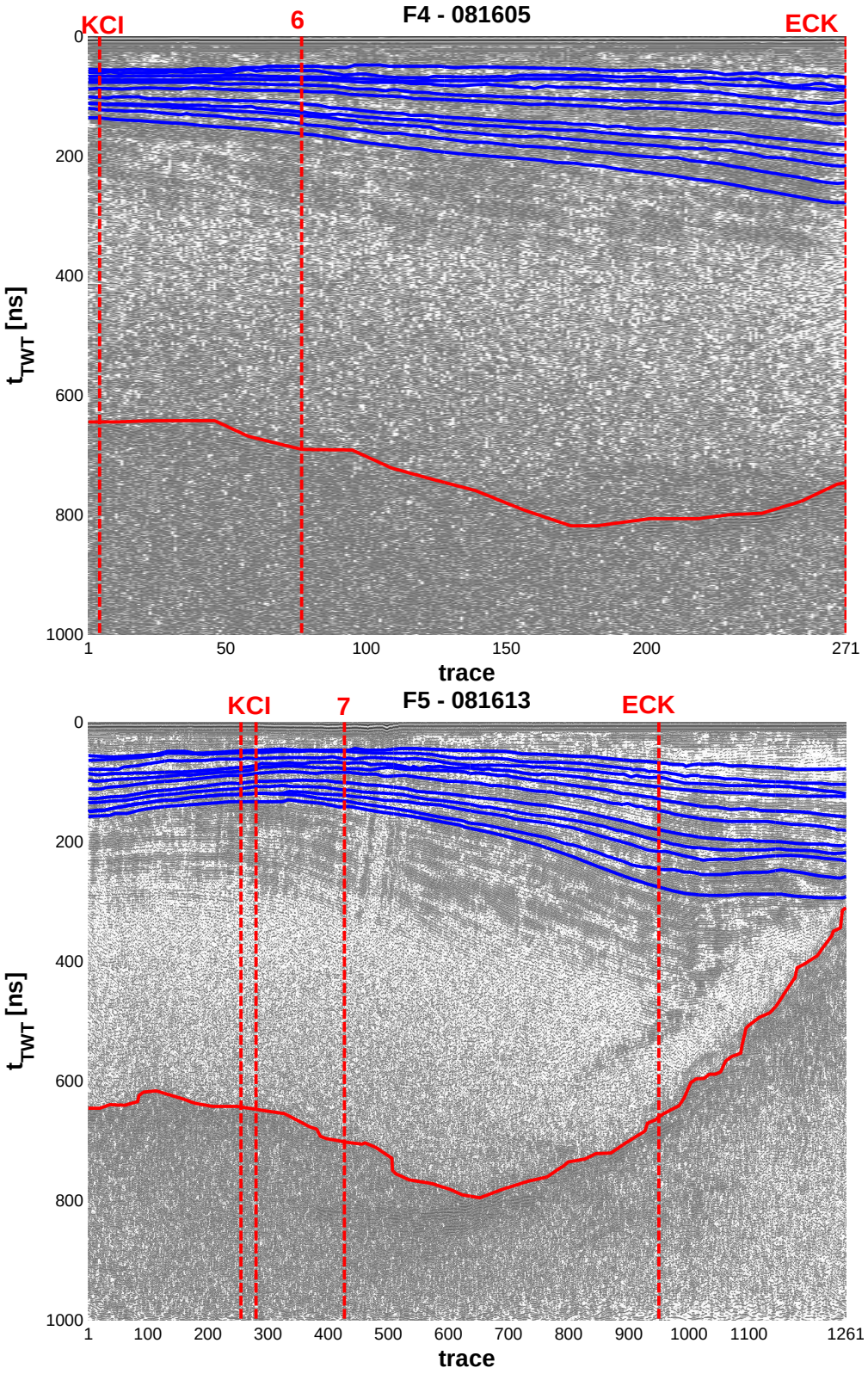


Figure B.4
Radargram of F4 and F5.

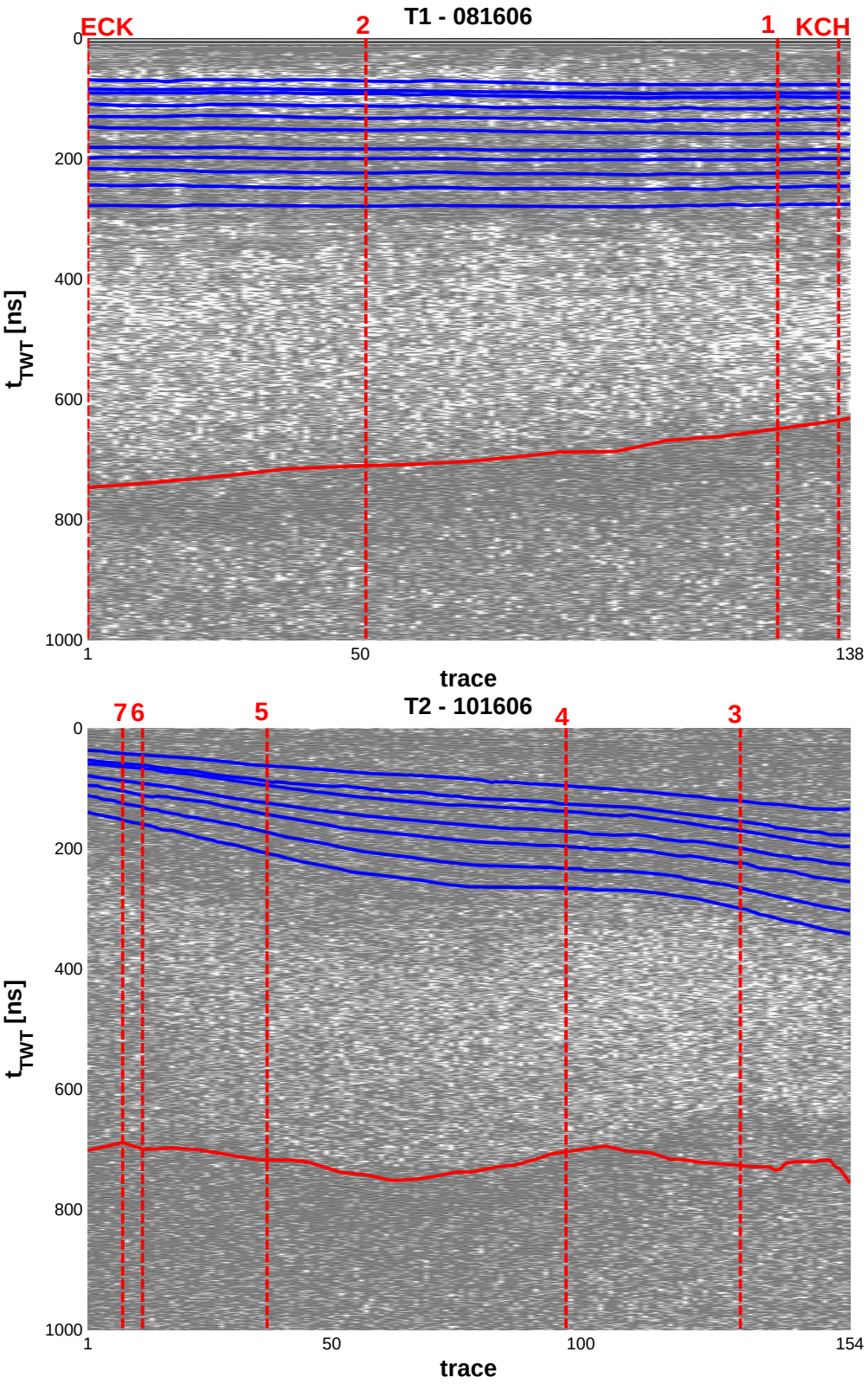


Figure B.5
Radargram of T1 and T2; the poor resolution is due to the small number of traces.

B.3 Error propagation in TWT processing and IRH dating

This section deals with the mathematical details of the error calculation with respect to ice thickness, depth of internal reflection horizons and IRH dating. In this context, the wave speed error has to be discussed as well.

Wave speed

$$c(\rho(z)) = \frac{c_0}{1 + 0.845\frac{cm^3}{g} \cdot \rho(z)} \qquad \text{(cf. equations (4.3) and (4.4))}$$

$$\Rightarrow \quad \Delta c(\rho(z)) = \frac{c_0 \cdot 0.845\frac{cm^3}{g}}{\left(1 + 0.845\frac{cm^3}{g} \cdot \rho(z)\right)^2} \cdot \Delta\rho(z) \tag{B.1}$$

where $\Delta\rho$ is composite of the smoothing error and the interpolation error as explained in section 5.2.

Ice thickness

Here, c and Δc are functions of the relative depth Σ, not of absolute depth z.

$$H = \frac{t^H_{\mathrm{TWT}}}{2} \left[\underbrace{\int_0^1 \frac{1}{c(\rho(\Sigma))}\,\mathrm{d}\Sigma}_{F} \right]^{-1} \qquad \text{(cf. equation (4.6))}$$

There are two independent error sources: $\Delta t^H_{\mathrm{TWT}}$ and $\Delta c(\rho(\Sigma))$.

$$\Delta H_{\mathrm{TWT}} = \frac{\Delta t^H_{\mathrm{TWT}}}{2} \int_0^1 \frac{1}{c(\rho(\Sigma))}\,\mathrm{d}\Sigma \qquad \left(\Delta t^H_{\mathrm{TWT}} = 20ns\right)$$

$$\Delta H_c = \frac{t^H_{\mathrm{TWT}}}{2} \cdot \frac{\Delta F}{F^2}$$

$$\text{where } \Delta F = \int_0^1 \frac{\Delta c(\rho(\Sigma))}{c(\rho(\Sigma))^2}\,\mathrm{d}\Sigma \overset{\text{eq. (B.1)}}{=} \frac{1}{c_0} \int_0^1 0.845\frac{cm^3}{g} \cdot \Delta\rho(\Sigma)\,\mathrm{d}\Sigma$$

$$\Delta H = \sqrt{(\Delta H_{\mathrm{TWT}})^2 + (\Delta H_c)^2}$$

Depth of internal reflection horizons

Here, c and Δc are functions of z. The IRH depth z^{IRH} is not calculated explicitly (as for instance H) but implicitly:

$$t_{\mathrm{TWT}} = 2 \int_0^z \frac{1}{c(\rho(\tilde{z}))}\,\mathrm{d}\tilde{z}\ .$$

The integration is carried out until $t_{\mathrm{TWT}} = t_{\mathrm{TWT}}^{\mathrm{IRH}}$. Then it is $z^{\mathrm{IRH}} = z$. The problem of error propagation is solved by rearranging this relation as follows:

$$t_{\mathrm{TWT}}^{\mathrm{IRH}} = 2\, z^{\mathrm{IRH}}\, F' \quad \Leftrightarrow \quad z^{\mathrm{IRH}} = \frac{t_{\mathrm{TWT}}^{\mathrm{IRH}}}{2\, F'} \quad \text{where } F' = \frac{1}{z^{\mathrm{IRH}}} \int_0^{z^{\mathrm{IRH}}} \frac{1}{c(\rho(\tilde{z}))}\,\mathrm{d}\tilde{z}\ .$$

The IRH depth z^{IRH} is assumed to be without uncertainty in the expression for F'. Otherwise, the error must be calculated by iteration due to the implicit calculation. Δz^{IRH} is calculated in accordance with the error of ice thickness:

$$\Delta z_{\mathrm{TWT}}^{\mathrm{IRH}} = \frac{\Delta t_{\mathrm{TWT}}^{\mathrm{IRH}}}{2\, F'} \qquad (\Delta t_{\mathrm{TWT}}^{\mathrm{IRH}} = 5ns)$$

$$\Delta z_c^{\mathrm{IRH}} = \frac{t_{\mathrm{TWT}}^{\mathrm{IRH}}}{2}\,\frac{\Delta F'}{F'^2} \qquad \left(\Delta F' = \frac{1}{c_0}\int_0^{z^{\mathrm{IRH}}} 0.845\frac{cm^3}{g}\cdot\Delta\rho(\tilde{z})\,\mathrm{d}\tilde{z}\right)$$

$$\Delta z^{\mathrm{IRH}} = \sqrt{(\Delta z_{\mathrm{TWT}}^{\mathrm{IRH}})^2 + (\Delta z_c^{\mathrm{IRH}})^2}$$

IRH age

The IRH dating error results from

- dating error of the cores: $\Delta t_{\mathrm{dat}}^{\mathrm{dat}} = 3$ a in the relevant time interval ($t_{\mathrm{dat}}(z)$ is the core dating).
- depth error of the IRH: $\Delta t_z^{\mathrm{dat}} = \left.\frac{\mathrm{d}t_{\mathrm{dat}}(z)}{\mathrm{d}z}\right|_{z=z^{\mathrm{IRH}}} \Delta z^{\mathrm{IRH}}$

$$\Rightarrow \Delta t^{\mathrm{dat}} = \sqrt{(\Delta t_{\mathrm{dat}}^{\mathrm{dat}})^2 + (\Delta t_z^{\mathrm{dat}})^2}$$

C Appendix to chapter 5

C.1 Accumulation rate distributions from GPR

The accumulation rate along the GPR profiles are necessary as model input (in the case of profiles which are parallel to flow lines) and as data set for spatial interpolation (which is also related to model input).

The accumulation rate at each trace of a GPR profile is calculated by assuming constant vertical strain (Nye's model – section 3.2) to account for annual layer thinning. Equation (3.8) is solved for $\dot{b}$:

$$\dot{b} = -\frac{H}{t} \cdot \ln\left(1 - \frac{z}{H}\right) \ .$$

The depth of the considered IRH and the ice thickness at the specific trace are inserted in meter water equivalent. Furthermore, t is set the respective IRH age according to Table 4.1.

As mentioned in section 5.1, the calculated patterns of accumulation rate from younger horizons are higher than those from the older ones.

The accumulation rate distribution along T2 is rejected in the data set for spatial interpolation because of the lack of confidence in the IRH picking and the resulting incoherent course of the accumulation rates: at small trace numbers, these are unexpectedly close too each other and at high trace numbers they differ strongly.

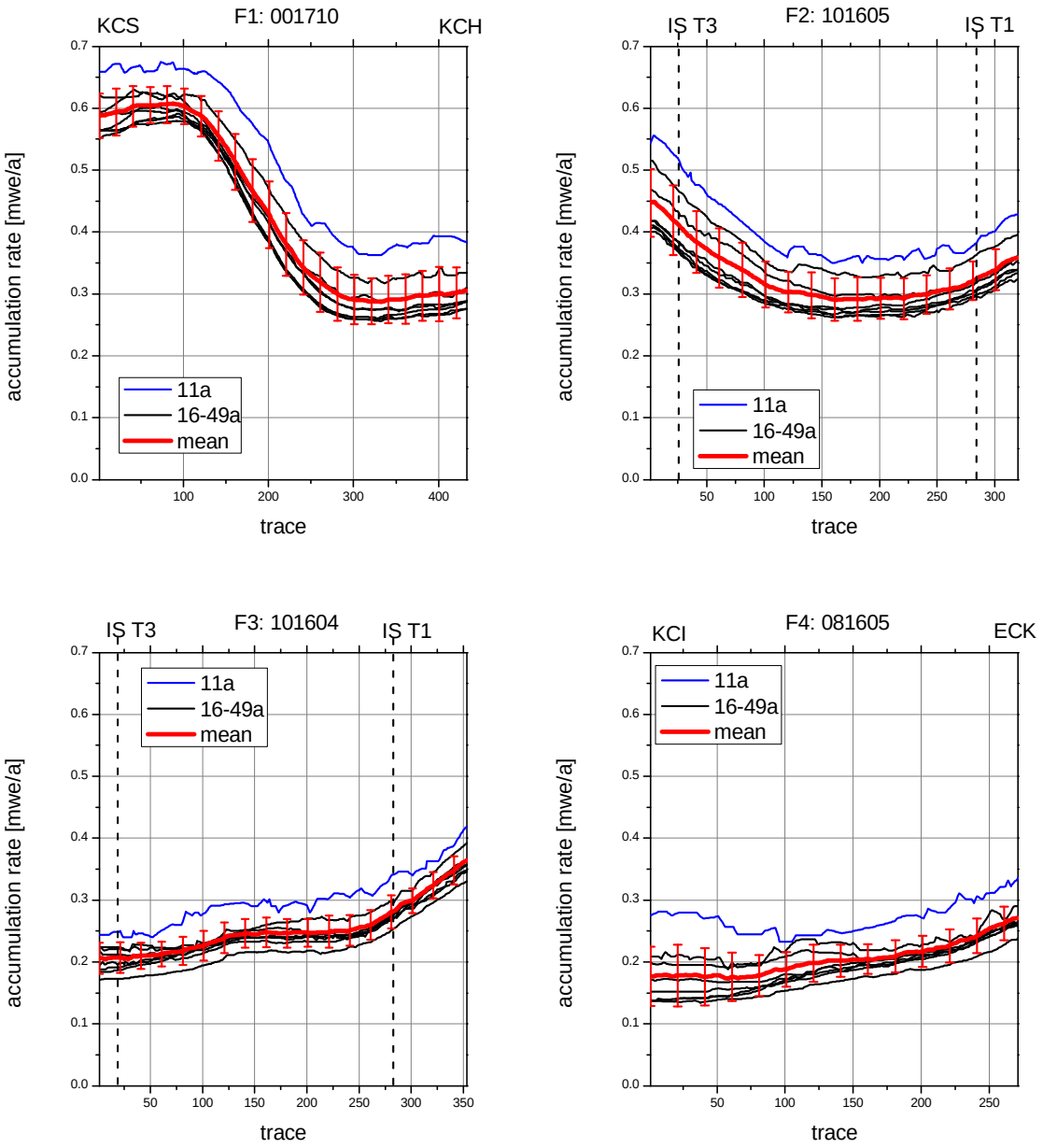


Figure C.1
Accumulation Rate Distributions from GPR 1; the blue and black lines are those calculated from the IRHs (ages are listed in the legend). The red lines are the mean rates including the blue distribution with the according error of the mean.

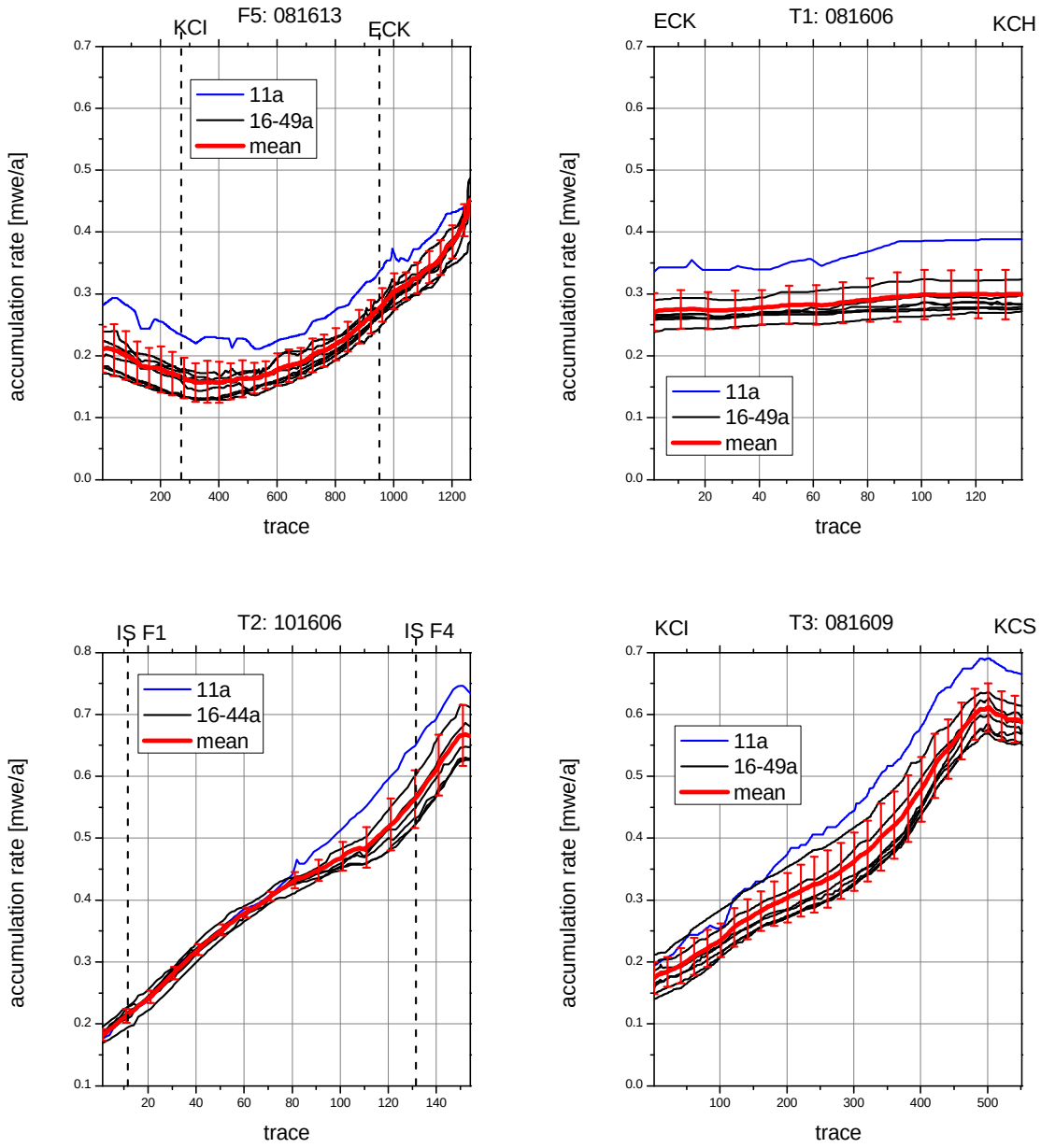


Figure C.2
Accumulation Rate Distributions from GPR 2; the blue and black lines are those calculated from the IRHs (ages are listed in the legend). The red lines are the mean rates including the blue distribution with the according error of the mean.

C.2 Estimation of density interpolation error at CC

The comparison of the density distribution of the ice core CC to the corresponding density distribution according to the interpolation scheme defined in sections 5.2 (KCH- and KCS-density as input density at the flow line's edges) and 3.5.2 (distance-weight interpolation between KCH and KCS) serves as basis for the estimation of density interpolation error.

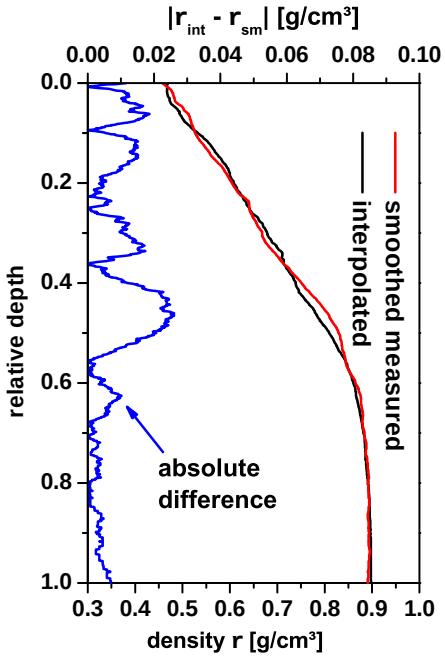


Figure C.3
Interpolated density at CC vs. smoothened measured density at CC; the interpolated density is obtained according to the scheme outlined in section 5.2. The maximum interpolation error $\Delta\rho_{\text{int}} = 0.02$ g/cm^3 is estimated from the blue curve in this plot.

C.3 Surface velocity results from optimization

Here, results of the partially rejected method of fixing SYNDICATE's flux parameters q_0 and D by fitting them to measured surface velocities (section 5.3) are shown: The surface velocity is calculated by

$$u_S(x_0) = \frac{5}{4}\,\frac{\rho_{\text{liquid}}}{\rho(x_0, Z=0)}\,\frac{1}{H(x_0)}(1-D)\left(\int_0^{x_0} \dot{b}(x_0)\,\mathrm{d}\tilde{x}_0 + q_0\right)$$

with q_0 and D obtained from the optimization of the linear system (5.2).

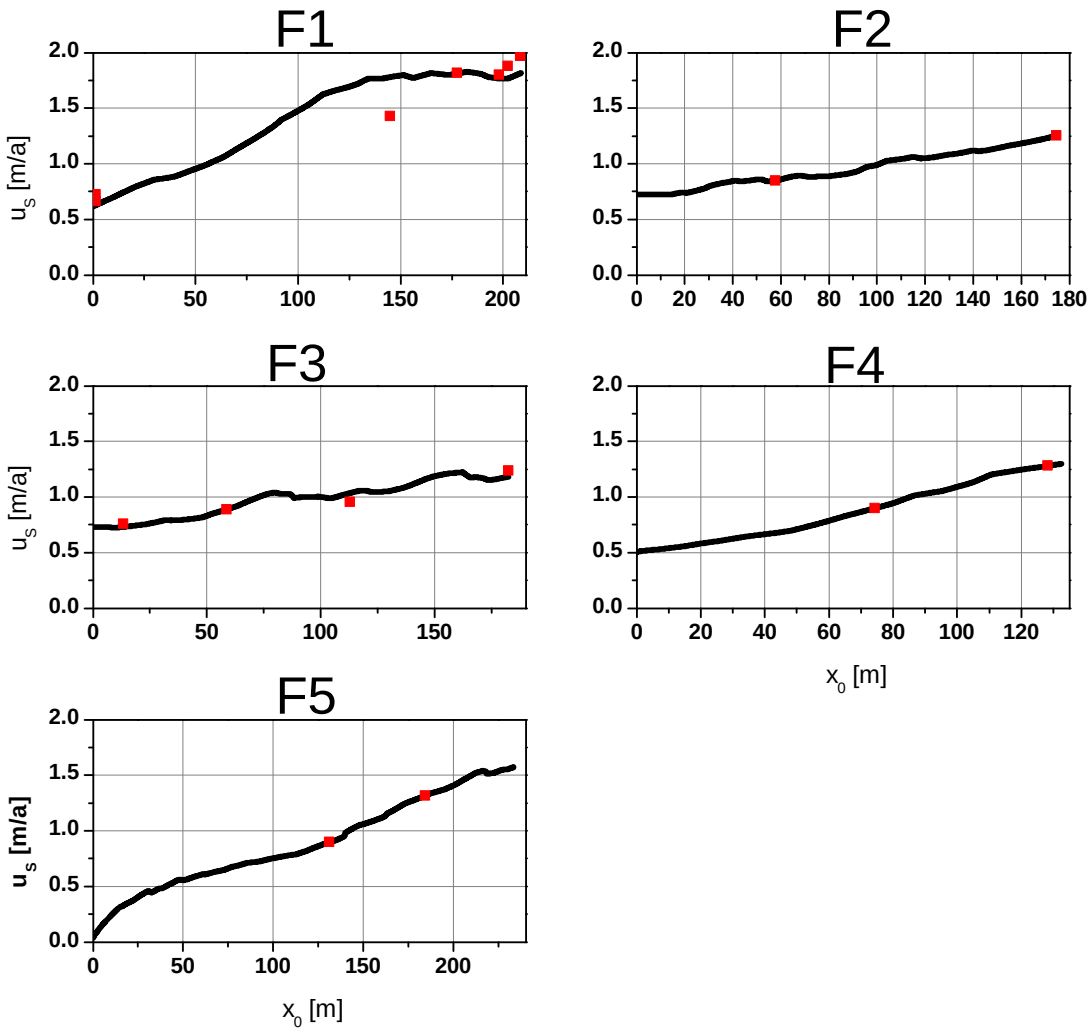


Figure C.4
Calculated (black lines) and measured (red markers) surface velocities; the calculated ones are obtained from the optimization explained in section 5.3. The measured ones were allocated to the specific flow lines by eye (Figure 5.4).

flow line	F1	F2	F3	F4	F5
q_0^{opt} [m mwe/a]	21.5	50.7	47.6	21.9	0.6
D^{opt}	0.39	0.71	0.68	0.42	0.35

Table C.1
Flux parameters from optimization; the red D-entries are the basis for the final choice of D-values on the flow lines.

C.4 Kriging

In this section, a brief introduction to kriging is given. Then, the kriging performance with respect to the usage in this thesis is discussed (variogram analysis, some results which were not shown in section 5.4, interpolation error estimation).

Introduction to kriging

A random function Z (here: surface altitude, bedrock altitude and accumulation rate) is considered in a certain area. It is sampled at the location $\vec{r}_1$, ..., $\vec{r}_N$: $Z(\vec{r}_i) = Z_i$ [Chilès and Delfiner, 1999].[15] At a certain location $\vec{r}_0$, $Z(\vec{r}_0)$ is estimated by

$$Z^*(\vec{r}_0) = \sum_{i=1}^{N} w_i \, Z_i \ . \tag{C.1}$$

The weights w_i depend on the relative position of $\vec{r}_0$ and $\vec{r}_i$. In the case of kriging, it is demanded that Z^* meets the following requirements: The estimation is linear, unbiased and best (error minimized) [Kitanidis, 1997]:

1. **linear**: equation (C.1).

2. **unbiased**: the estimation error $Z^*(\vec{r}_0) - Z(\vec{r}_0)$ is zero on average.

$$E\left[Z^*(\vec{r}_0) - Z(\vec{r}_0)\right] = 0 \overset{\text{eq. (C.1)}}{=} E\left[\sum_{i=1}^{N} w_i \, Z_i - Z(\vec{r}_0)\right] \ . \tag{C.2}$$

3. **best**: the estimation variance is minimized:

$$\sigma_E^2 = E\left[\left(Z^*(\vec{r}_0) - Z(\vec{r}_0)\right)^2\right] \overset{!}{=} \min \ . \tag{C.3}$$

There are mainly three types of kriging, namely

- simple kriging: the mean m of Z in the considered area is constant and known.
- ordinary kriging: m is constant but not known.
- universal kriging: m is assumed to follow a trend.

[15] Here: $\vec{r}_i = (Y_i, X_i)^{\mathrm{T}}$ are two-dimensional vectors given in the Swiss grid.

In this thesis, ordinary kriging is performed (cf. p. 122 – Variogram fitting and kriging performance). A constant mean m results in $E\left[Z(\vec{r}_0)\right] = m$ and $E\left[Z_i\right] = m$. According to equation (C.2), this leads to

$$\sum_{i=1}^{N} w_i = 1 \ . \tag{C.4}$$

The covariance function C of the random variable Z is

$$C(\vec{h}) = E\left[\Big(Z(\vec{r}+\vec{h}) - m\Big) \cdot \big(Z(\vec{r}) - m\big)\right]$$

where $\vec{h}$ is the lag vector between two points in the considered area. C does not depend on these two points (here: $\vec{r}$ and $\vec{r}+\vec{h}$), but only on their relative position. The estimation variance (equation (C.3)) is expressed by C [Wackernagel, 1995]:

$$\begin{aligned}
\sigma_E^2 &= E\left[\left(Z^*(\vec{r}_0) - Z(\vec{r}_0)\right)^2\right] \\
&= E\left[Z^*(\vec{r}_0)^2 + Z(\vec{r}_0)^2 - 2\, Z^*(\vec{r}_0)\, Z(\vec{r}_0)\right] \\
&= \sum_{i=1}^{N}\sum_{j=1}^{N} w_i\, w_j\, C(\vec{r}_i - \vec{r}_j) + \underbrace{C(\vec{r}_0 - \vec{r}_0)}_{C(\vec{0})} - 2\sum_{i=1}^{N} w_i\, C(\vec{r}_i - \vec{r}_0) \ .
\end{aligned}$$

In ordinary kriging $C(\vec{h})$ is mostly substituted by the variogram

$$\gamma(\vec{h}) = \frac{1}{2} E\left[\Big(Z(\vec{r}+\vec{h}) - Z(\vec{r})\Big)^2\right] = C(\vec{0}) - C(\vec{h}) \ . \tag{C.5}$$

Note that the reversal $C(\vec{h}) = C(\vec{0}) - \gamma(\vec{h})$ is not possible in general, i.e. from each covariance the variogram can be derived but not vice versa. The variogram values $\gamma(\vec{h})$ are low for high correlation of Z at two points $\vec{r}+\vec{h}$ and $\vec{r}$ and vice versa. The kriging variance expressed by γ is

$$\sigma_E^2 = -\gamma(\vec{0}) - \sum_{i=1}^{N}\sum_{j=1}^{N} w_i\, w_j\, \gamma(\vec{r}_i - \vec{r}_j) + 2\sum_{i=1}^{N} w_i\, \gamma(\vec{r}_i - \vec{r}_0) \ . \tag{C.6}$$

The kriging variance σ_E^2 is minimized with respect to the weights w_i under the constraint of equation (C.4). This is equivalent to solving the linear system in equation (C.7). With the solution w_1^{s}, ..., w_N^{s}, the estimated value $Z^*(\vec{r}_0)$ is determined:

$$\begin{pmatrix}
\gamma(\vec{r}_1 - \vec{r}_1) & \cdots & \gamma(\vec{r}_1 - \vec{r}_N) & 1 \\
\vdots & \ddots & \vdots & \vdots \\
\gamma(\vec{r}_N - \vec{r}_1) & \cdots & \gamma(\vec{r}_N - \vec{r}_N) & 1 \\
1 & \cdots & 1 & 0
\end{pmatrix} \cdot \begin{pmatrix} w_i \\ \vdots \\ w_N \\ m \end{pmatrix} = \begin{pmatrix} \gamma(\vec{r}_1 - \vec{r}_0) \\ \vdots \\ \gamma(\vec{r}_N - \vec{r}_0) \\ 1 \end{pmatrix} \ . \tag{C.7}$$

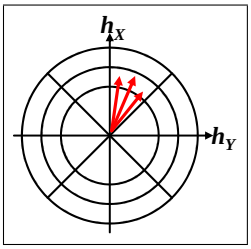


Figure C.5
Classification of lag vectors: The vectors are allocated to the segments. In this sense, the illustrated vectors (red) are within one class.

At this stage, an assumption with respect to the spatial correlation of Z has to be made, namely: $\gamma(\vec{h})$ has to be specified. This is done via the experimental variogram: The values of the sample set $\{(\vec{r}_i,\ Z_i)\}$ are checked for their spatial correlation by pairs: The lag vectors $\vec{h}_{ij} = \vec{r}_j - \vec{r}_i$ are classified with respect to their directions and lengths as it is shown in Figure C.5. Then the experimental variogram for one class $\vec{h}_p$ (p specifies the lag vector class, i.e. the segment) is

$$\gamma_{\exp}(\vec{h}_p) = \frac{1}{2\,N_p} \sum_{\substack{\vec{h}_{ij} \text{ in} \\ \text{class } p}} \left(Z(\vec{r}_i) - Z(\underbrace{\vec{r}_i + \vec{h}_{ij}}_{=\vec{r}_j}) \right)^2 .$$

N_p is the number of lag vectors $\vec{h}_{ij}$ in the class p.

The theoretical variogram $\gamma(\vec{h})$ is fitted to these data $\{(\vec{h}_p,\ \gamma_{\exp}(\vec{h}_p))\}$ for application in equation (C.7). Typical models are polynomials, gaussian distributions and exponential functions. Note, that $\gamma(\vec{0}) \neq 0$ is an option in kriging. If this is the case (nugget effect), the estimator is not exact any more ($Z^*(\vec{r}_i) \neq Z_i$), but smoothing. Since spatial correlation can depend on the direction of $\vec{h}$, further parameters for the variogram fit which model the anisotropy are introduced [Wackernagel, 1995]. Their meaning is shown in Figure C.6.

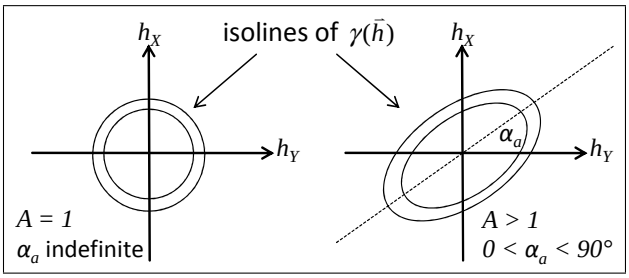


Figure C.6
Anisotropy parameters in variogram fitting; the isolines of the variogram are elliptic. The quantity A is the ratio of the lengths of major axis and minor axis. The angle α_a defines the direction of the major axis (if $A > 1$) or of the minor one (if $A < 1$).

Variogram fitting and kriging performance

The data sets are detrended before the experimental variogram is calculated since a trend in the input data strongly influences the correlation of the samples. Barnes [2003] recommends this procedure instead of performing universal kriging, which would be necessary since the interpolated quantities clearly exhibit trends. The trend is a polynomial of Y and X (Swiss grid) of second degree. The variogram models, which are fit to the data, are a nugget effect γ_n and a 'cubic' model which is basically a polynomial of third degree featuring a minimum $\gamma_c(h = 0) = 0$ where $h = |\vec{h}|$ and a maximum $\gamma_c(h = h_l) = \gamma_l$ (index l means length). At $h > h_l$, the variogram is constant. Then it is $\gamma(\vec{h}) = \gamma_n + \gamma_c(\vec{h})$ and the fit parameters are γ_n, γ_l and h_l. The anisotropy parameters, namely the anisotropy angle α_a and the anisotropy ratio A are further fit parameters. Their effect is illustrated in Figure C.6.

The specific parameter fitting is constrained by the following considerations:

- The anistropy ratio was restricted to values $A \leq 2$ according to Barnes [2003] ($A = 3$ is recommended as maximum, but $A = 2$ is the default maximum of the software). This has led to mismatch of the theoretical variogram in some directions. An attempt to increase A to avoid this mismatch led to extreme ridges in the interpolated surfaces and was thus rejected.
- The nugget γ_n is forced to values greater than $\sim$0.1% of the experimental variogram's maximum for smoothing purpose.

The variograms obtained from the data sets of surface altiude, ice thickness and accumulation rate are shown in the following (Figures C.7, C.8 and C.9). The extraordinarily uneven spacing of data points (high data density along the GPR profiles but large gaps between them) leads to difficulties in variogram fitting: The mentioned restriction of anisotropy ratios implies that there is mismatch between the experimental variograms and the fitted ones in some directions.

In interpolation, not every data point is considered for interpolation: The extension of the matrix in equation (C.7) is decreased by computing $Z^*(\vec{r}_0)$ using a maximum of 64 pairs $(\vec{r}_i, Z_i)$, which are chosen by their distance to $\vec{r}_0$.

Variograms

The variograms are each given in four directions: $\alpha = 0, 45, 90, 135° \pm 10°$. α is the angle to the Y-axis (Swiss grid). The black points belong to the experimental variogram. The blue lines are the modeled variogram.

Surface altitude

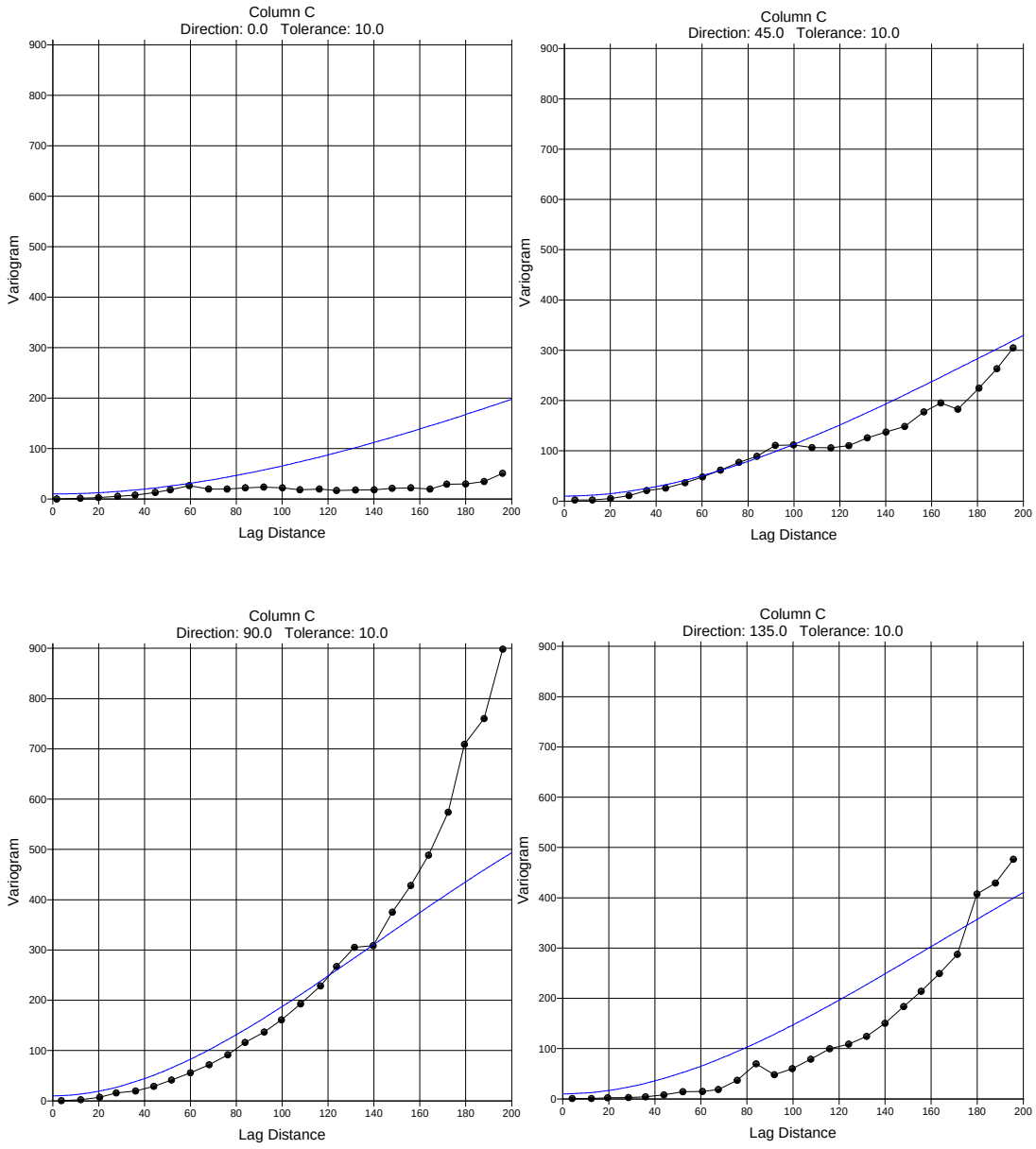


Figure C.7
Variogram of detrended surface altitude.

Ice thickness

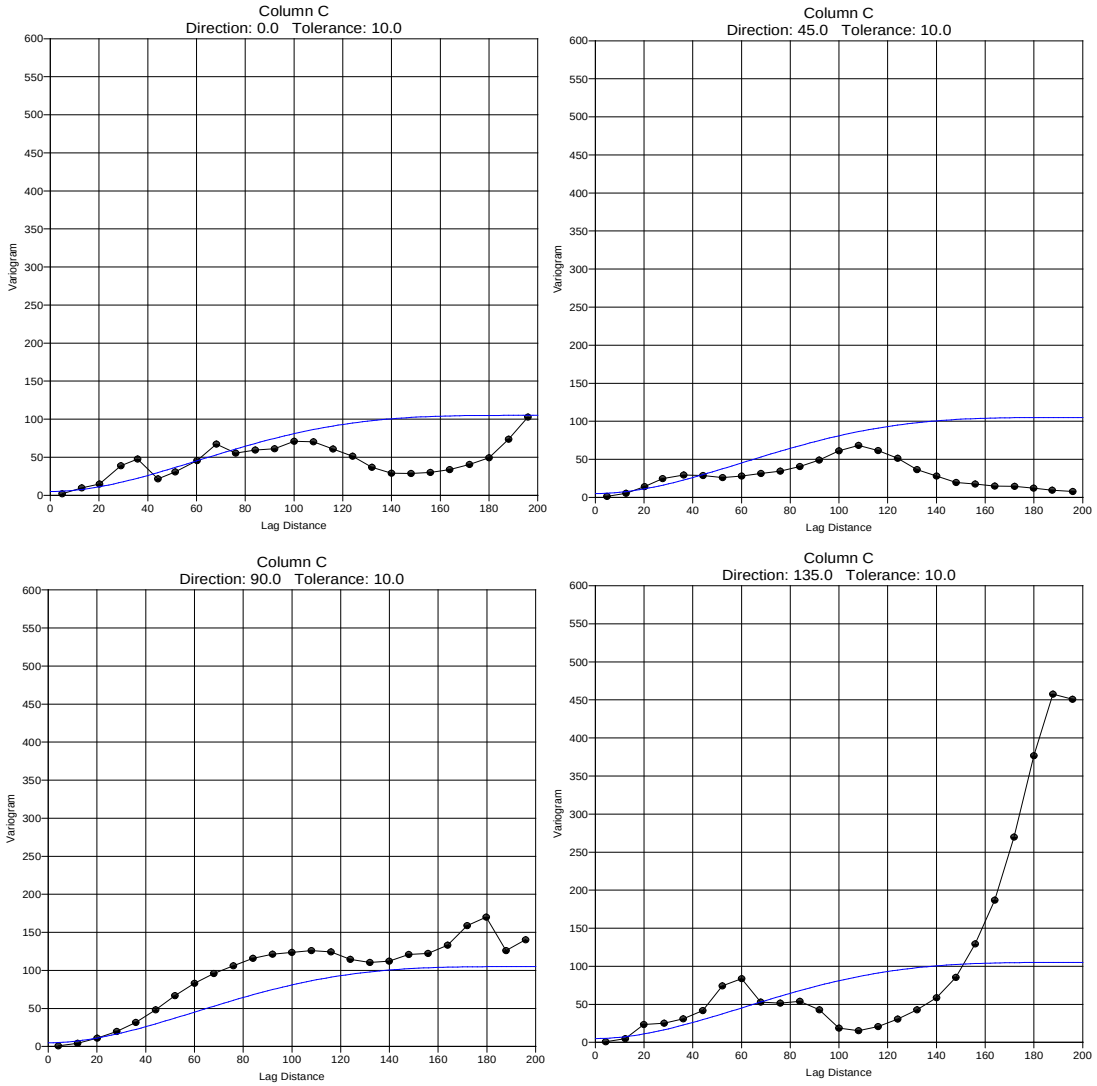


Figure C.8
Variogram of detrended ice thickness.

Accumulation rate

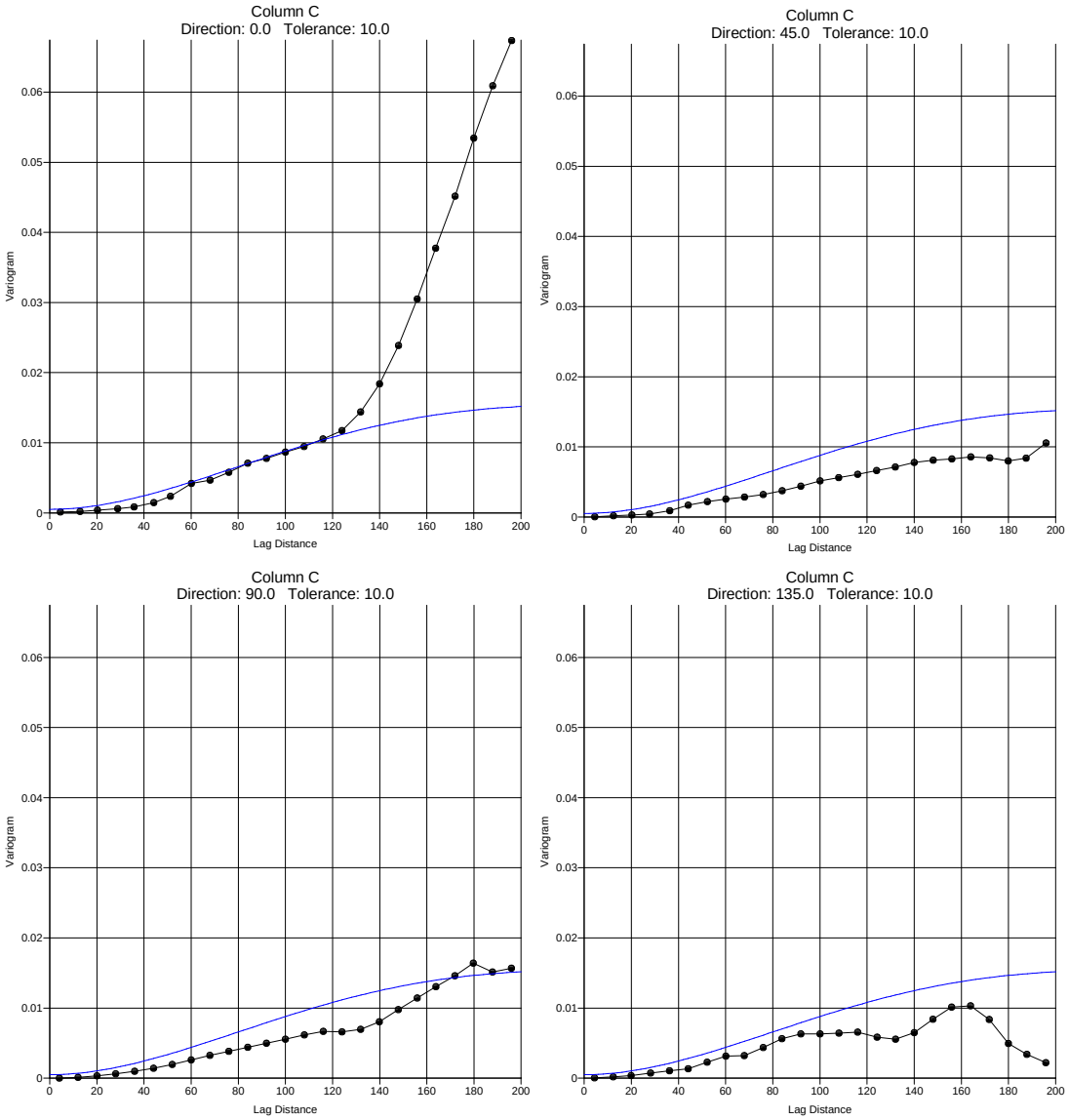


Figure C.9
Variogram of detrended accumulation rate.

Further kriging results

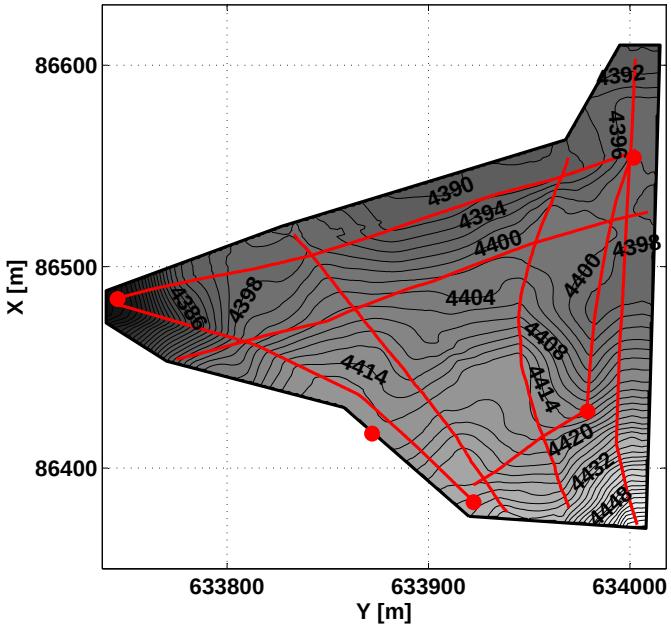


Figure C.10
Interpolated bedrock altitude; red lines: data points (GPR profiles); the bedrock altitude is obtained by subtracting the interpolated ice thickness from the interpolated surface altitude (Figures 5.6 and 5.7)

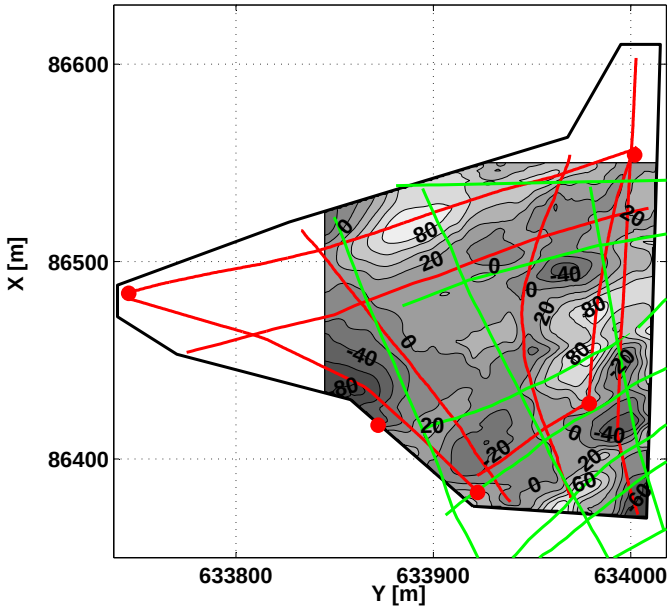


Figure C.11
Comparison of TWT of bedrock reflection: 2003; red: data points from 2008/10; green: data points from 2003 [Böhlert, 2005]; the illustrated TWT difference is $t_{\mathrm{TWT}}^{(2008/10)}(Y,X) - t_{\mathrm{TWT}}^{(2003)}(Y,X)$.

Interpolation error

Although the kriging error (equation (C.6)) is calculated inherently in kriging interpolation, the error of the interpolated quantities used for the bootstrapping is determined as follows (see section 5.4.1):

Several subsets of the data samples are left out. The interpolation is carried out with the remaining data points. In the case of ice thickness and accumulation rate, these subsets are the single GPR profiles. In the case of surface altitude, the subsets are defined by circles randomly generated within the area of interest (Figure C.12).

The quantities which are interpolated from the reduced samples are compared to the original values at the omitted data points. The difference is considered as a function the distance to the nearest data point which is not omitted (Figures C.13, C.14 and C.15). Then, the interpolation error is determined by a linear fit (parameters given in the figures) to these data, except for surface altitude.

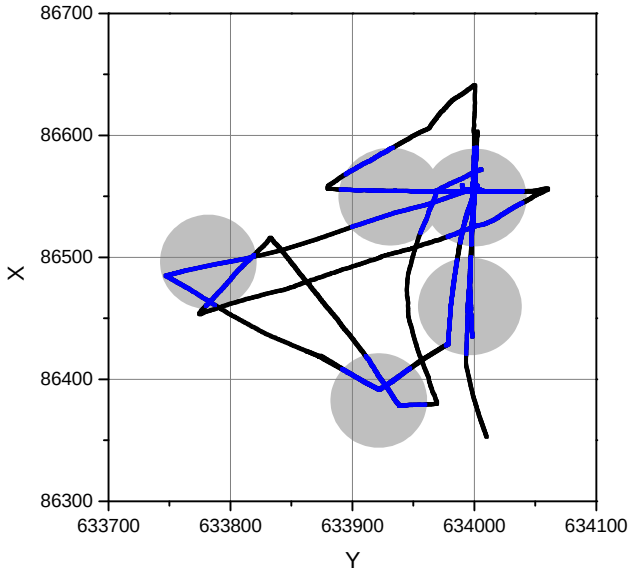


Figure C.12
Omitted data points for error estimation of surface altitude; each set is marked by blue dots and the corresponding including circle.

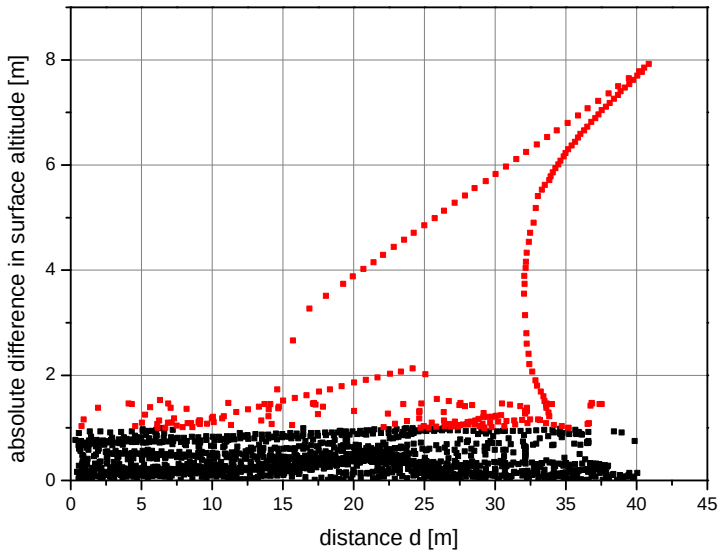


Figure C.13
Difference of surface altitude at omitted data points vs. distance to the nearest data point which was not left out; the red marked values are not taken into account due to their high variation. The extraordinarily high ones result from the omission of GPS data near KCS (cf. section 5.4.1).

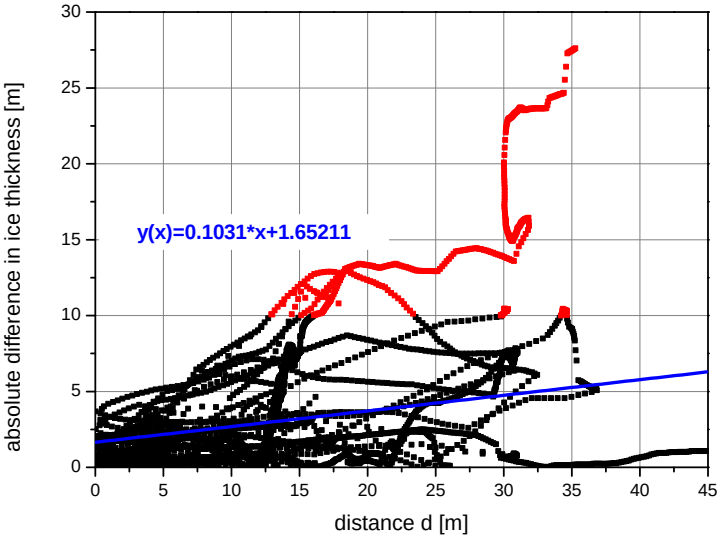


Figure C.14
Difference of ice thickness at omitted data points vs. distance to the next data point which was not left out; the red marked values are not taken into account due to their high variation.

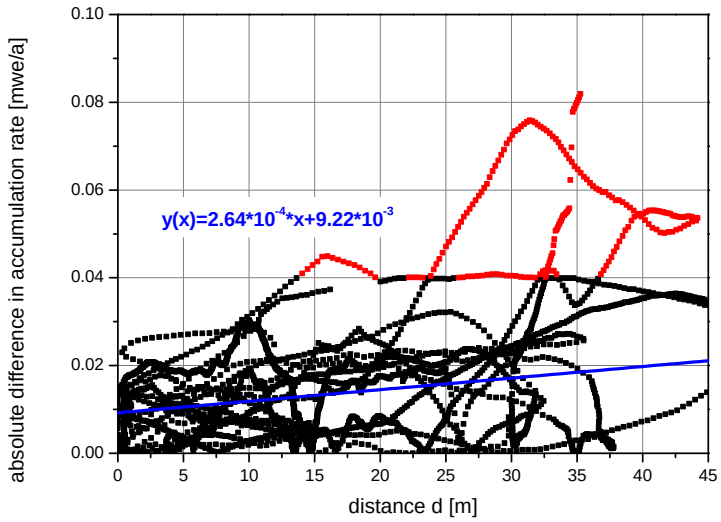


Figure C.15
Difference of accumulation rate at omitted data points vs. distance to the next data point which was not left out; the red marked values are not taken into account due to their extraordinary high variation.

D Appendix to chapter 6

D.1 Isochrone comparison GPR vs. SYNDICATE

In this section, the comparison of isochrones from GPR data to those obtained from SYNDICATE along the remaining primary flow lines (F2, F4, F5) is illustrated.

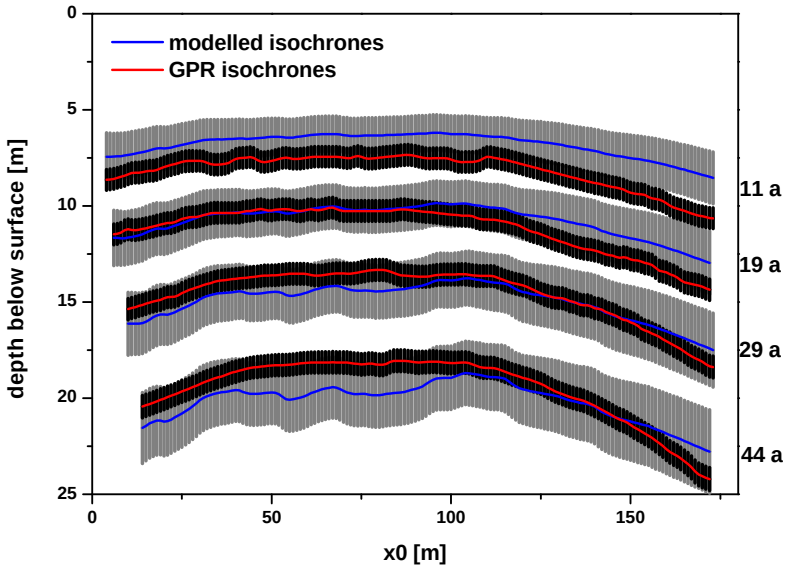


Figure D.1
Comparison of selected isochrones from GPR and SYNDICATE on F2; the grey bonds indicate the error of the modeled isochrones, the black ones those of IRHs. The ages of the illustrated IRHs and and model isochrones are on the right. The isochrones fit within the error ranges.

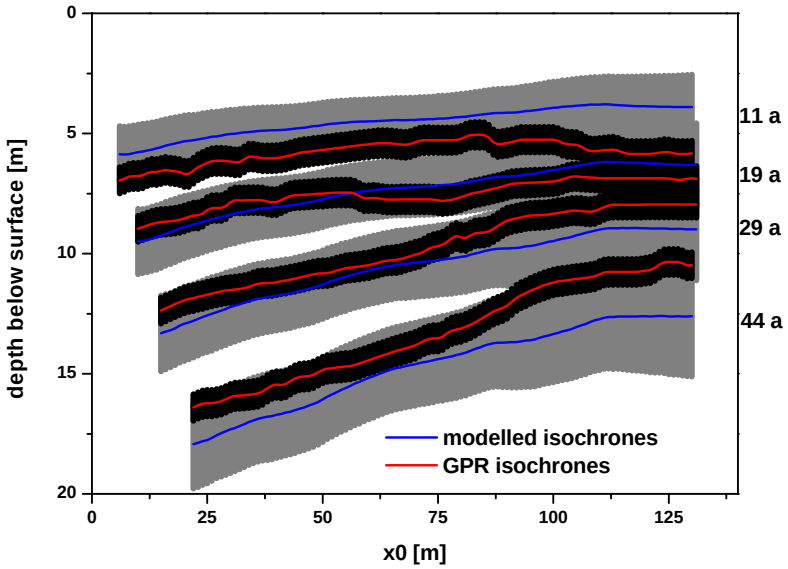


Figure D.2
Comparison of selected isochrones from GPR and SYNDICATE on F4; the grey bonds indicate the error of the modeled isochrones, the black ones those of IRHs. The ages of the illustrated IRHs and and model isochrones are on the right. The isochrones fit within the error ranges, but the systematic course of the GPR isochrones is reproduced poorly by the model.

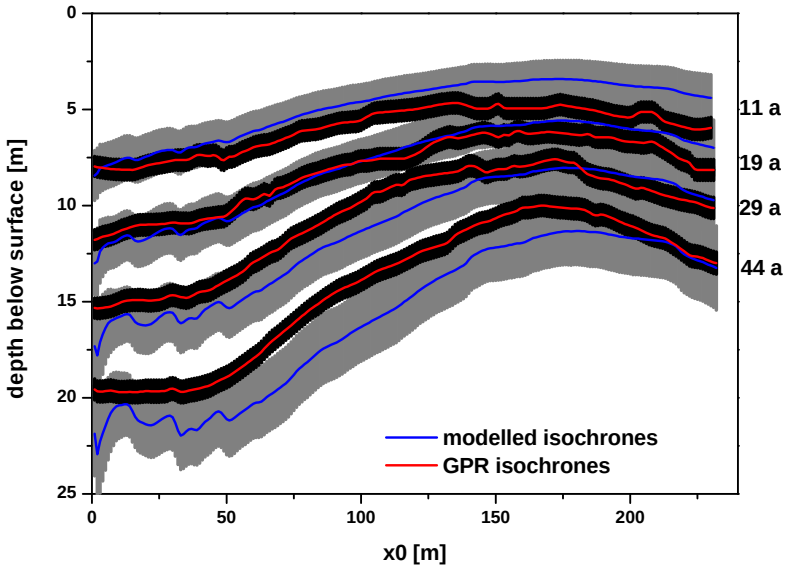


Figure D.3
Comparison of selected isochrones from GPR and SYNDICATE on F5; the grey bonds indicate the error of the modeled isochrones, the black ones those of IRHs. The ages of the illustrated IRHs and and model isochrones are on the right. Note the oscillations of the modeled isochrones at lower x_0. They most likely result from the steep bedrock topograpy in this area (cf. appendix D.3). The isochrones fit within the error ranges partially except for that of age $t = 44$ a where there are some regions of mismatch (e.g. at 50 m $< x_0 <$ 100 m).

D.2 Source regions KCS

According to the original objectives outlined in chapter 1, the source regions of the ice in the cores is of great importance with respect to ice core dating. Here, the results concerning the saddle core KCS are illustrated.

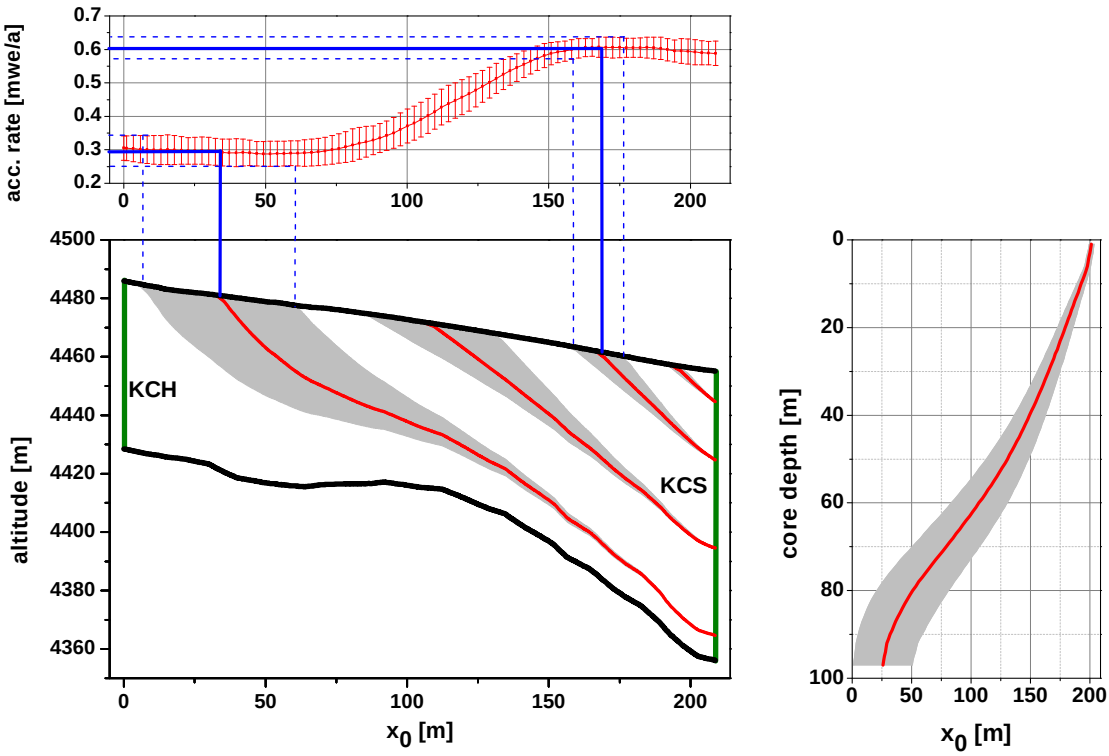


Figure D.4
Left bottom: exemplary trajectories (red) and corresponding uncertainty (grey) leading to specific depth at the KCS location; right bottom: distribution of core depth of KCS vs. location of accumulation on F1; top: accumulation rate distribution along F1; a possible usage is illustrated by the blue lines (dashed lines: corresponding uncertainty): The layer in a specific core depth can be allocated to the region where it was deposited. At the corresponding x_0-coordinate, the accumulation rate can be evaluated and therefore be allocated to the core depth, where the consideration started.

D.3 Model runs on full length vs. shortened flow line on F5

Since the interpolation of model isochrones revealed artefacts near KCI, the output along the single flow lines were checked. On F5, the source of this error was found (Figure D.5). A shortened version of F5 does not show the inconsistency (Figure D.5 as well). Consequently, F5 is shortened for further evaluation.

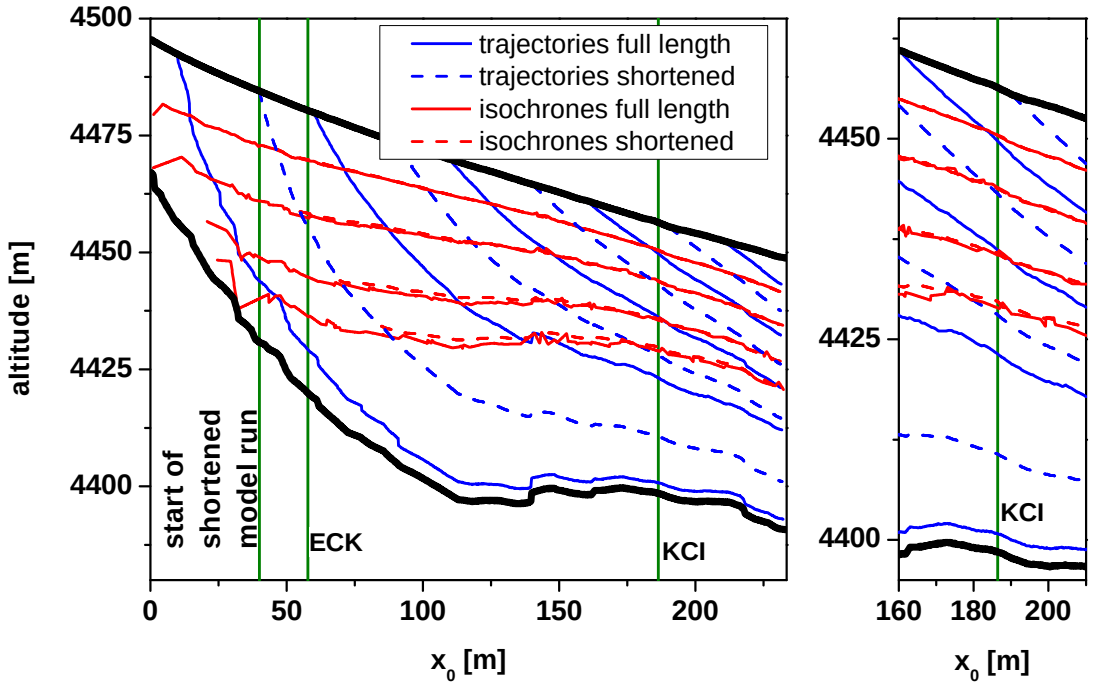


Figure D.5
Model run on F5: full length vs. shortened flow line; note the unreasonable isochrone run ('oscillations'). The illustrated isochrones are those of t = 20, 50, 100, 150 a. Right: zoom to KCI; The unreasonable isochrone run affects the isochrone structure here as well. This explains the artifacts in the top of Figure 6.9 and is most likely a consequence of the steep bedrock at low x_0-values in combination with the relatively high accumulation rate. Note than 'shortened' and 'full length' in the legend refer to the flow line, not to the specific trajectories and isochrones.

D.4 Variograms of isochrone interpolation

In the following, the variograms, from which the isochronous surfaces discussed in sections 6.3.1 and 6.3.2 are interpolated, are presented, namely the model isochrones of age t = 40-150 a with an increment of 10 a in the case of the two data sets which differ only in the length of the primary flow line F5 (see appendix D.3): set 1 designates the set in which F5 is contained in full length and which features the inconsistency near KCI (Figure 6.9); accordingly, set 2 contains the shortened version of F5.

As those variograms used in chapter 5 and presented in appendix C.4, the variograms here are obtained by first detrending the respective data sets. In the case of set 1, the applied model is linear and features a nugget effect which, however, turns out to be close to zero in all fits. Set 2 is interpolated by applying the cubic model and the nugget effect as described in appendix C.4. (The different models are not the cause for the different interpolated surfaces and the inconsistency at KCI.)

The variograms are designated by the corresponding model ages. On the abscissa, there is the lag distance (appendix C.4) and on the ordinate the variogram values. The compact illustration in combination with the possibilities of the software result in the poor quality with respect to axes designation. All illustrated variograms in appendix D.4 are shown for one angle and 90° tolerance and thus include all directions within the horizontal plane. This the reason for the apparent mismatch of experimental and theoretical variogram in some cases, e.g. the variogram of the 110-years-isochrone of set 2.

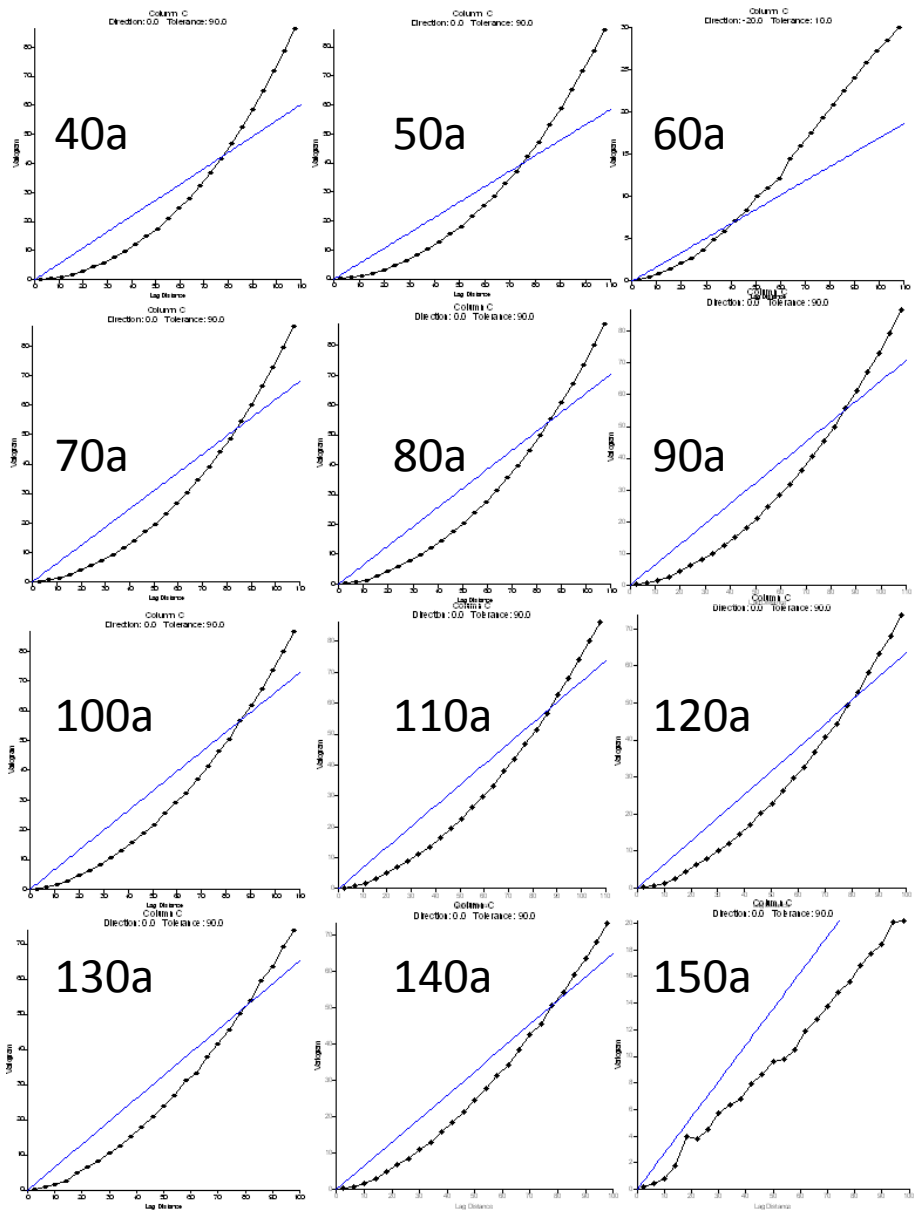


Figure D.6
Variograms of detrended interpolated isochrones from all flow lines (F5 in full length – set 1).

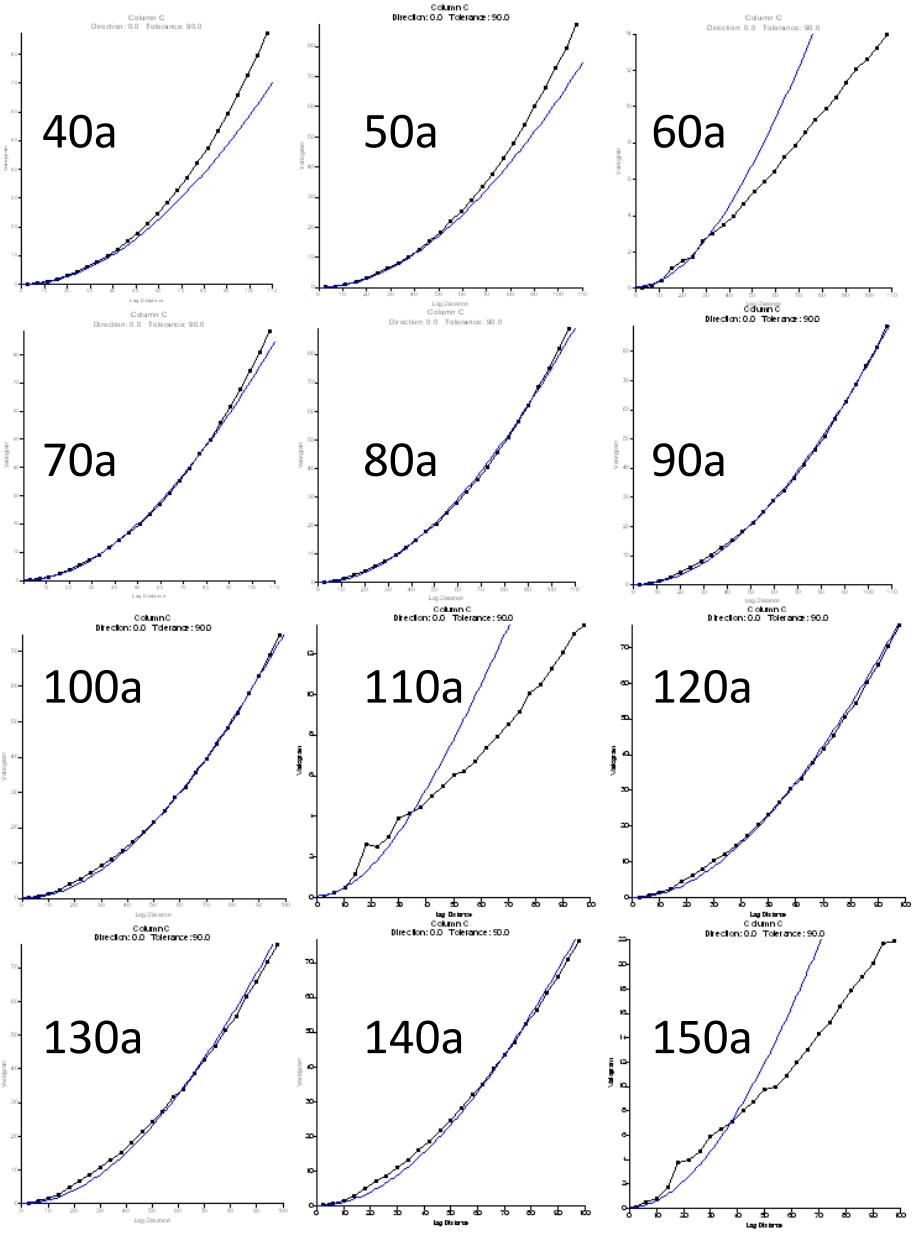


Figure D.7
Variograms of detrended interpolated isochrones (F5 shortened – set 2); here the variogram models are the same as presented in appendix C.4.

D.5 Interpolated isochrones and data points

Here, the absence of data points for isochrone interpolation at KCH and CC is illustrated: In contrast to KCS and KCI, there are no data points directly at CC (it is offside the model flow line and GPR profile F1 by a few meters) and at KCH, where the distance to they the data points increases with increasing model (and core) age. This is accompanied by a decrease of confidence in the interpolated values at the slope cores.

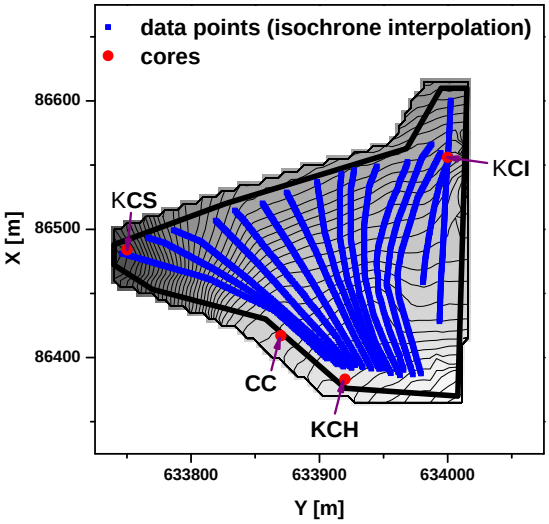


Figure D.8
Interpolated isochrone ($t = 60$ a).

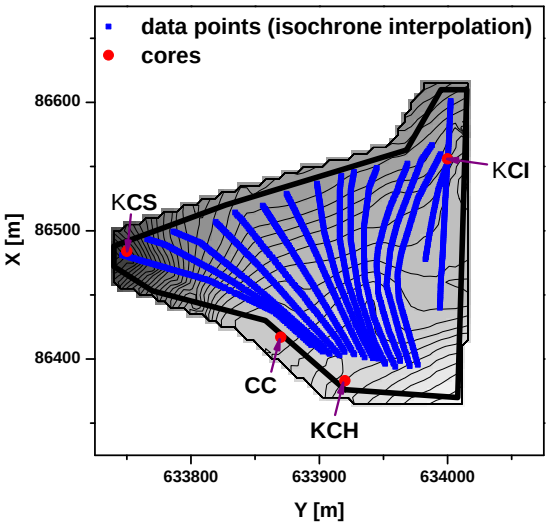


Figure D.9
Interpolated isochrone ($t = 100$ a)

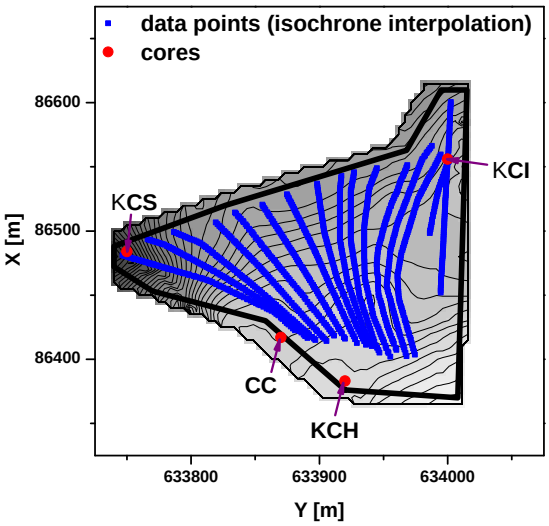


Figure D.10
Interpolated isochrone ($t = 140$ a)

E Code

```
c     Syndicate
c     letzte wesentliche Aenderungen: 29.10.2010
c     letzte Aenderung an der Dokumentation: 03.03.2011

      integer i,i0,j0,i1,j1,n,np,npb,test,n1,n2,i3,i4
      integer i2,ioper,zus,nfs,numxinterp,numzinterp,numflow
      double precision z,href,dhdx,zwe,sigstep,HA,HE
      double precision x0,pashor,pasver,pascou,q,q0
      double precision umoy,usurf,h,k,uz,w,ureal,wreal
      double precision hmoy,hsuiv,z0,x1,teta,pi,dist,dz1,dz2
      double precision couche,wprece,zprece,initia,temps
      double precision diverg,z0prec,zf1,zf2,tf1,tf2,qr
      integer iA,iE
      double precision xA,xE
c     Variablen aus Inputdateien/Interpolation
      double precision, dimension(:), allocatable :: z1,z2,x,bil,xb
      double precision xfs(150),zfs(150),tempsfs(150)
      double precision, dimension(:), allocatable :: zp1,zp2
      double precision, dimension(:), allocatable :: xpb,bilp
      double precision, dimension(:), allocatable :: qb
      double precision, dimension(:), allocatable :: h1,hwe
      double precision, dimension(:), allocatable :: xd,zd,d,d0,dEnde
      double precision, dimension(:), allocatable :: sig0,sigEnde
      double precision, dimension(:), allocatable :: sigma
      double precision, dimension(:), allocatable :: dsig0,dsigEnde
      double precision, dimension(:), allocatable :: dsig(:,:)
      character f1*14,f2*14,f3*14,f4*14

      pi=acos(0.)*2.

c     die Datei response muss enthalten:
c     1. Pfad zu Datei mit Geomtrie-Input
c     2. Pfad zu Datei mit Akk.raten-Input
c     3. Pfad zu Datei mit Dichte-Input
c     4. Pfad zu Datei mit Startpunkten fuer zusaetzliche Trajektorien
c     die Datei response2 muss enthalten (zeilenweise)
c     1. pashor=horizontale Schrittweite bei der Interpolation
c     2. pascou=Abstand der Fließlinienstartpunkte an der OF
c     3. dist=x-Wert des Bohrlochs
c     4. diverg=Divergenz (zwischen 0 und 1)
c     5. q0=Eisfluss von links (in m*mwe/a)
      open(8,file='response',status='old')
      open(9,file='response2',status='old')

      ioper=1
      read(8,*)f1
      read(8,*)f2
      read(8,*)f3
      read(8,*)f4
      read(9,*)pashor
      pasver=pashor
c     (pasver=vertikale Schrittweite bei der I. (fuer Dichte))
      read(9,*)pascou
      read(9,*)dist
      read(9,*)diverg
      read(9,*)q0
      open(10,file=f1,status='old')
      open(11,file=f2,status='old')
      open(12,file=f3,status='old')
      open(13,file=f4,status='old')
      open(14,file='resul')
      open(16,file='resul2')
      open(17,file='resul3')
      open(21,file='geom_interpol')
      open(23,file='massb_interpol')
      open(25,file='Integer')

      i=1
c     Einlesen der Geometrie
100   read(10,*,end=105)a,b,c
        i=i+1
      goto 100
105   close(10)
      open(10,file=f1,status='old')

      allocate(x(i))
      allocate(z1(i))
      allocate(z2(i))
      write(25,9020)i

      i=1
110   read(10,*,end=120)x(i),z1(i),z2(i)
      i=i+1
      goto 110
120   n=i-1

      numxinterp=nint((x(n)-x(1))/pashor)+1
      write(25,9020)numxinterp
c     numxinterp=Anzahl Punkte in x-Richtung
c     fuer alle folgenden Interpolationen
      i=1
      allocate(h1(numxinterp))
      allocate(hwe(numxinterp))
      allocate(xpb(numxinterp))
      allocate(zp1(numxinterp))
      allocate(zp2(numxinterp))
      allocate(bilp(numxinterp))
      allocate(qb(numxinterp))
c     ---------------------------------------------------------------
c     Interpolation der Hoehe des Bettes z1 und der OF z2
c     bei jedem horizontalen Schritt
c     x0,zp1(.),zp2(.)
c     ---------------------------------------------------------------
      i1=1
      x0=0
      zp1(1)=z1(1)
      zp2(1)=z2(1)

c     beschreibe geom_interpol
      write(21,9011)x0,zp1(1),zp2(1)

130   x0=x0+pashor
      if (x0.gt.x(n))goto 150
        i1=i1+1
      do 135 i=1,n
        if(x(i+1).gt.x0.and.x(i).lt.x0)then
          zp1(i1)=((z1(i+1)-z1(i))/(x(i+1)-x(i)))*(x0-x(i))+z1(i)
          zp2(i1)=((z2(i+1)-z2(i))/(x(i+1)-x(i)))*(x0-x(i))+z2(i)
        endif

c     Falls man mit x0 näher als pashor/10 an einer vorgegebenem x-Wert des
c     Geometrie-Files liegt, werden die entsprechenden Hoehen uebernommen
        if(abs(x(i+1)-x0).lt.pashor/10.)then
          zp1(i1)=z1(i+1)
          zp2(i1)=z2(i+1)
        endif
        if(abs(x(i)-x0).lt.pashor/10.)then
          zp1(i1)=z1(i)
          zp2(i1)=z2(i)
        endif

135   continue

c     beschreibe geom_interpol
      write(21,9011)x0,zp1(i1),zp2(i1)
      goto 130
c     Anzahl der interpolierten Punkte
150   np=i1
c     Schliesse geom_interpol
      close(21)

c     finde dickste Stelle und berechne hieraus die Anzahl der z-Punkte
c     im Gitter
151   do 152 i=1,numxinterp
        h1(i)=zp2(i)-zp1(i)
152   continue
153   href=maxval(h1)
      numzinterp=nint(href/pasver)+1
      write(25,9020)numzinterp
c     numzinterp=Anzahl von z-Punkten (auf relativer Skala)
c     Ende Geometrie-Input
c     ---------------------------------------------------------------
c     Einlesen der Massenbilanz
      i=1
```

```
155   read(11,*,end=157)a,b
      i=i+1
      goto 155
157   close(11)
      open(11,file=f2,status='old')
      allocate(bil(i))
      allocate(xb(i))
      write(25,9020)i
      i=1
160   read(11,*,end=170)xb(i),bil(i)
        i=i+1
      goto 160
170   n=i-1
c     bil muss in mwe/a gegeben sein.
c     ---------------------------------------------------------------
c     Interpolation der Massenbilanz
c     s.o.
c     Ergebnis: xpb(.),bilp(.)
c     ---------------------------------------------------------------
      i1=1
      x0=0
      xpb(1)=xb(1)
      bilp(1)=bil(1)
c     Beschreibe massb_interpol
      write(23,9013)x0,bilp(1)
180   x0=x0+pashor
      if (x0.gt.xb(n))goto 200
        i1=i1+1
        xpb(i1)=x0
      do 185 i=1,n
        if(xb(i+1).gt.x0.and.xb(i).lt.x0)then
          bilp(i1)=((bil(i+1)-bil(i))/(xb(i+1)-xb(i)))
     1*(x0-xb(i))+bil(i)
        endif
c     Falls x0 näher als pashor/10 an einem geg. x-Wert liegt, wird
c     der entsprechende Wert uebernommen
        if(abs(xb(i+1)-x0).lt.pashor/10.)then
          bilp(i1)=bil(i+1)
        endif
        if(abs(xb(i)-x0).lt.pashor/10.)then
          bilp(i1)=bil(i)
        endif
185   continue
c     Beschreibe massb_interpol
      write(23,9013)x0,bilp(i1)
      goto 180
c     Anzahl der interpolierten Punkte
200   npb=i1
c     Schliesse massb_interpol
      close(23)
c     Ende Massenbilanz-Input
c     -----------------------------------------------
c     qr ist der letzte x-Wert... Die Bezeichnung ist zufaellig
c     (zu viele Variablennamen mit 'x' schon vergeben!)
      qr=xpb(npb)
c     ---------------------------------------------------
c     Einlesen der Dichte
c     x-Wert xd()   Tiefe zd()   Dichte d()
c
c     Bsp.:
c     0   0    0.35
c     0   2    0.40
c     0   10   0.6
c     0   40   0.9
c     0    0    0
c     100  0    0.35
c     ....etc
c
c     Zwischen die beiden Profile muessen 3 Nullen gesetzt werden
c     ebenso am Ende der Datei
c     ---------------------------------------------------
      i=1
      test=0
205   read(12,*,end=210)a,b,c
      if(c.lt.0.01.and.test.eq.0)then
        n1=i-1
        test=1
      endif
      i=i+1
      goto 205

210   close(12)
      open(12,file=f3,status='old')
      n2=i-2
      allocate(xd(i-1))
      allocate(zd(i-1))
      allocate(d(i-1))
      allocate(sig0(n1))
      allocate(d0(n1))
      allocate(sigEnde(n2-n1-1))
      allocate(dEnde(n2-n1-1))

      write(25,9020)n1
      write(25,9020)n2
      i1=1
      i=1

214   read(12,*,end=215)xd(i),zd(i),d(i)
        i=i+1
      goto 214
c     Dichte d an den beiden Endpunkten wird festgesetzt,
c     und zwar in relativer Tiefe sig zwischen 0 und 1
c     Finde x-Werte der beiden Dichte-Punkte für Tiefenumrechnung
215   do 216 i=1,numxinterp
        if(abs(xpb(i)-xd(1)).lt.pashor/1.9)then
          HA=h1(i)
        endif
        if(abs(xpb(i)-xd(n1+2)).lt.pashor/1.9)then
          HE=h1(i)
        endif
216   continue

217   do 218 i=1,n1
        sig0(i)=zd(i)/HA
        d0(i)=d(i)
218   continue

      do 219 i=n1+2,n2
        sigEnde(i-n1-1)=zd(i)/HE
        dEnde(i-n1-1)=d(i)
219   continue

      allocate(sigma(numzinterp))
      allocate(dsig0(numzinterp))
      allocate(dsigEnde(numzinterp))

      sigstep=1./(dble(numzinterp)-1.)

      do 220 i=1,numzinterp
        sigma(i)=(dble(i)-1.)*sigstep
220   continue

c     Interpoliere in sigma-Richtung an den beiden Endpunkte
c     es besteht nun also der Zusammenhang
c     sigma(i),dsig0(i),dsigEnde(i)
      do 230 i=1,numzinterp

        do 222 i1=1,n1-1
          if(sigma(i).gt.sig0(i1).and.sigma(i).lt.sig0(i1+1))then
            dsig0(i)=(d0(i1)-d0(i1+1))/(sig0(i1)-sig0(i1+1))
     1        *(sigma(i)-sig0(i1))+d0(i1)
          endif
          if(abs(sigma(i)-sig0(i1)).lt.sigstep/10.)then
            dsig0(i)=d0(i1)
          endif
          if(abs(sigma(i)-sig0(i1+1)).lt.sigstep/10.)then
            dsig0(i)=d0(i1+1)
          endif
c     ersetze den Rest durch 0.9
          if(sigma(i).gt.sig0(n1))then
            dsig0(i)=0.9
          endif
222     continue
        do 225 i1=1,n2-n1-2
```

```
           if(sigma(i).gt.sigEnde(i1).and.sigma(i).lt.sigEnde(i1+1))then
              dsigEnde(i)=(dEnde(i1)-dEnde(i1+1))/
     1         (sigEnde(i1)-sigEnde(i1+1))
     1          *(sigma(i)-sigEnde(i1))+dEnde(i1)
           endif
           if(abs(sigma(i)-sigEnde(i1)).lt.sigstep/10.)then
             dsigEnde(i)=dEnde(i1)
           endif
           if(abs(sigma(i)-sigEnde(i1+1)).lt.sigstep/10.)then
             dsigEnde(i)=dEnde(i1+1)
           endif
c     ersetze den Rest durch 0.9
           if(sigma(i).gt.sigEnde(n2-n1-1))then
             dsigEnde(i)=0.9
           endif
225        continue
230   continue
c     Interpoliere dsig in xRichtung
c     also dsig = Dichte auf dem x-sigma-Gitter
      allocate(dsig(numxinterp,numzinterp))

c     falls Dichte nicht an den Endpunkten gegeben:
c     andere Interpolation:
      if(abs(xd(1)).gt.pashor)goto 1000
      if(abs(xd(n1+2)-xpb(numxinterp)).gt.pashor)goto 1000

      do 231 i=1,numzinterp
         dsig(1,i)=dsig0(i)
         dsig(numxinterp,i)=dsigEnde(i)
231   continue

      do 240 i3=2,numxinterp-1
         do 235 i=1,numzinterp
            dsig(i3,i)=(dsigEnde(i)-dsig0(i))/(xpb(numxinterp)-xpb(1))
     1      *(xpb(i3)-xpb(1))+dsig0(i)
235        continue
240   continue
c     ----------------------
c     Einlesen der zusätzlichen Startpunkte fuer Fliesslinien
c     Bezeichnungen selbsterklaerend
c     Falls keine vorhanden, muss die Datei einfach '0  0  0' enthalten.
c     ----------------------
299   i=1
300   read(13,*,end=310)xfs(i),zfs(i),tempsfs(i)
      i=i+1
      goto 300
310   nfs=i-1
c     --------------------
c     Berechnung der Eisdicken in mwe (Integral)
c     --------------------
      do 315 i=1,numxinterp
         hwe(i)=0.
         do 313 i3=1,numzinterp-1
            hwe(i)=hwe(i)+(dsig(i,i3)+dsig(i,i3+1))/2.*sigstep
313        continue
           hwe(i)=h1(i)*hwe(i)
315   continue
c     ------------ ---------------------------------------
c     Flussberechnung unter Beruecksichtigung der Divergenz
c     (Integral ueber die Akkumulationsrate)
c     xpb(.),qb(.)
c     ------------------------------------------------------
      do 320 i=1,npb
        q=q0
        do 330 j=1,i
          q=q+bilp(j)*pashor
330     continue
        qb(i)=q*(1.-diverg)
320   continue

c     numflow=Nummer der Fliesslinie
      numflow=0
c     Berechnung der Fliesslinien
c     test=1 bedeutet: zusaetzliche Fliesslinien aus File
c     test=0 bedeutet: Fliesslinie ab Oberflaeche

      test=1
      zus=1
c     falls keine zusaetzlichen Fliesslinien vorhanden
c     (0,0,0 in der entsprechenden Datei):
c     starte direkt mit den Fliesslinien ab Oberflaeche
      if(xfs(1).eq.0..and.zfs(1).eq.0..and.tempsfs(1).eq.0.)then
        goto 349
      endif

335   if(zus.gt.nfs)goto 349
c     x0=Fliesslinienstartpunkt (im folgenden immer)
      x0=xfs(zus)
c     x1=x-Koordinate des aktuellen Standortes
      x1=x0
c     z=Tiefe des aktuellen Standortes
      z=zfs(zus)
      temps=tempsfs(zus)
c     zus=Index der zusaetzlichen Fliesslinien-Startpunkte
      zus=zus+1
      numflow=numflow+1
      goto 415
349   test=0
350   x0=0.-pascou
400   z=0.
      x0=x0+pascou
      numflow=numflow+1
      if(x0.gt.xpb(np))goto 600
      x1=x0
      temps=0.
      if(x1.gt.xpb(np))goto 400

      do 405 i=1,npb
        if(abs(x1-xpb(i)).lt.pashor/1.9)i0=i
405   continue
c     an der OF: Jahresschichtdicke=Akk.rate
      couche=bilp(i0)
c     Ebenso die Vertikalgeschw.
      wprece=bilp(i0)
      zprece=0.
      z0prec=zp2(i0)
      initia=bilp(i0)

c     Suche nach dem nächsten x-Wert des Gitters
c     dabei wird per unten definiertem Verfahren der "linke" naechste
c     Wert praeferiert, falls man (fast) genau zwischen zweien liegt.
415   do 416 i=1,npb
        if(abs(x1-xpb(i)).lt.pashor/1.9)i0=i
416   continue

c     h=Gletscherdicke an x1
      h=h1(i0)

c     Suche nach dem naechsten sigma-Wert des Gitters
c     wird benoetigt fuer Konversion der Geschwindigkeiten
c     von mwe/a zu m/a und fuer die Konversion von z in m zu z in mwe
      do 417 j=1,numzinterp
        if(abs(z/h-sigma(j)).lt.sigstep/1.9)j0=j
417   continue

c     Berechnung der Tiefe z in mwe (zwe)
      zwe=0.
      if(j0.eq.1)goto 419
      do 418 j=1,j0-1
        zwe=zwe+(dsig(i0,j)+dsig(i0,j+1))/2.*sigstep
418   continue
      zwe=h1(i0)*zwe

c     absolute Hoehe am Standort = Oberflaeche - Tiefe
419   z0=zp2(i0)-z

c     mittlere Geschwindigkeit(x1)=Fluss(x1)/Dicke(x1)
c     (parallel zur Oberflaeche)
      umoy=qb(i0)/h
c     Berechnung der Oberflaechengeschwindigkeit aus der
c     mittleren Geschwindigkeit (jeweils die Komponente || OF)
c     kein Gleiten!
      usurf=umoy/0.8
c     Ausgangspunkt: uz=k*zwe**4/4
```

```
c     an der OF: usurf=k*hwe**4/4
c     also gilt fuer k: k=(4*umoy/0.8)/hwe**4
      k=(4*umoy/0.8)/hwe(i0)**4
      uz=(hwe(i0)**4-zwe**4)*k/4.
c     Geschwindigkeit senkrecht zur Oberflaeche
      w=1.-(0.2*(zwe**4.)/(hwe(i0)**4.))
      w=bilp(i0)-1.25*bilp(i0)*(zwe/hwe(i0))*w
c     Umrechnung von mwe/a auf m/a:
      uz=uz/dsig(i0,j0)
      w=w/dsig(i0,j0)

c     Berechne Neigung der OF
c     teta=Neigungswinkel der Oberflaeche
c     teta > 0: rechts=hangabwaerts
c     teta < 0: rechts=hangaufwaerts
      if(i0.eq.npb)i2=i0-1
      if(i0.lt.npb)i2=i0+1
      teta=-1.*datan((zp2(i0)-zp2(i2))/(xpb(i0)-xpb(i2)))*200./pi
c     dhdx=Ableitung von h (Eisdicke)
c     nach x (fuer kinematische Korrektur)
c     dhwedx > 0: Dicke nimmt ab
c     dhwedx < 0: Dicke nimmt zu
      dhdx=-1.*(h1(i0)-h1(i2))/(xpb(i0)-xpb(i2))
c     kinematische Korrektur
      w=w-uz*zwe/hwe(i0)*dhdx

c     Drehe w und u in die abs. Vertikale/Horizontale
      wreal=-w*dcos(teta*pi/200.)-uz*dsin(teta*pi/200.)
      ureal=uz*dcos(teta*pi/200.)-w*dsin(teta*pi/200.)

c     Beschreibe resul
c     Inhalt:
c     x absolut
c     Hoehe absolut
c     horizontale (absolut) Geschwindigkeit
c     vertikale (absolut) Geschw.
c     Zeit
c     Nummer der Fliesslinie
      write(14,9004)x1,z0,ureal,wreal,temps,numflow

c     Beschreibe (evtl.) resul2 & resul3
c     Inhalt resul2:
c     abs. Hoehe, Tiefe, Alter, Quell-x
      if (x1.lt.dist.and.(x1+ureal).gt.dist)then
      write(16,9001)z0+(dist-x1)*wreal/ureal,z-(dist-x1)*wreal/ureal,
     1temps+(dist-x1)/ureal,x0
      endif

c     Inhalt resul3:
c     Tiefe,Alter,Jahresschichtdicke(mwe),JSD(m),JSD(OF),
      if (x1.lt.dist.and.(x1+ureal).gt.dist)then
      tf1=temps+(dist-x1)/ureal
      zf1=z+(dist-x1)*wreal/ureal
      couche=(zf1-zf2)/(tf1-tf2)
      write(17,9010)zf1,tf1,couche*dsig(i0,j0),
     1couche,initia
      tf2=tf1
      zf2=zf1
      endif
      wprece=w*dsig(i0,j0)

c     Endpunkt des Zeitschrittes:
      x1=x1+ureal
      z0=z0+wreal
      zprece=z
      z0prec=z0
c     Tiefe des Endpunktes
      z=z+w
      temps=temps+1.

c     Wenn man 1m vom Ende des Profils entfernt ist, wird die
c     Fliesslinie gewechselt
      if(test.eq.1.and.x1+1..gt.xpb(np))then
        goto 335
      endif
      if(x1+1..gt.xpb(np))then
        goto 400
      endif
c     Ebenso, wenn man 0.25m ueber dem Bett angekommen ist
      if(test.eq.1.and.z0.lt.zp1(i0)+0.25)then
        goto 335
      endif
      if(z0.lt.zp1(i0)+0.25)then
        goto 400
      endif

      goto 415

600   write(6,*)

      write(25,9020)numflow
800   goto 5000

c     ******************************************************
c     neue Dichte-Interpolation,
c     falls die gegebenen Profile nicht den Endpunkten entsprechen
1000  do 1010 i=1,numxinterp
         if(abs(xd(1)-xpb(i)).lt.pashor/1.9)then
           iA=i
           xA=xpb(i)
         endif
         if(abs(xd(n1+2)-xpb(i)).lt.pashor/1.9)then
           iE=i
           xE=xpb(i)
         endif
1010  continue
      do 1100 i=1,numzinterp
        dsig(iA,i)=dsig0(i)
        dsig(iE,i)=dsigEnde(i)
1100  continue
      do 1120 i3=iA+1,iE-1
         do 1110 i=1,numzinterp
            dsig(i3,i)=(dsigEnde(i)-dsig0(i))/(xE-xA)
     1      *(xpb(i3)-xA)+dsig0(i)
1110     continue
1120  continue

c     fliesslinienabwaerts wird die Dichte mit der Hoehe skaliert
c     Dies entspricht genau einer Gleichsetzung der dsig-Spalten
c     ab dem Index iE
      if(iE.lt.numxinterp)then
         do 1140 i3=iE+1,numxinterp
            do 1130 i=1,numzinterp
               dsig(i3,i)=dsig(iE,i)
1130        continue
1140     continue
      endif
c     fliesslinienaufwaerts: ebenso
      if(iA.gt.1)then
         do 1160 i3=1,iA-1
            do 1150 i=1,numzinterp
               dsig(i3,i)=dsig(iA,i)
1150        continue
1160     continue
      endif
      goto 299

5000  close(10)
      close(11)
      close(12)
      close(13)
      close(14)
      close(16)
      close(17)
      close(24)
      close(25)
9001  format(f9.4,2x,f9.6,2x,f9.4,2x,f9.4)
9004  format(f9.4,2x,f9.4,2x,f9.4,2x,f9.4,2x,f12.6,i4)
9010  format(f12.2,2x,f12.2,2x,f12.2,2x,f12.2,2x,f12.2)
9011  format(f12.4,2x,f12.4,2x,f12.4)
9013  format(f12.4,1x,f12.4)
9020  format(i5)
8000  stop
      end
```

```
function Bootstrap(Input,Durchlaeufe)
tic
% Lade Input-Parameter und deren Fehler
Input_Bootstrap(Input)
% Definiere globale Variablen aus Input-Datei
global xGeo OFGeo dOFGeo HGeo dHGeo xAkk bAkk dbAkk xDens1 zDens1 Dichte1 dDichte1
xDens2 zDens2 Dichte2 dDichte2 Divergenz dDivergenz q0 dq0 charAkk PfadArchiv
PfadCouran EndQuell
% Generiere File fuers Archiv (Divergenz und q0)
fid=fopen([PfadArchiv,'/input/Div_q0.txt'],'wt'); fclose(fid);
global durchlauf
h=waitbar(0,'Modell-Durchlaeufe');
for durchlauf=1:Durchlaeufe
    % Bereite die verschiedenen Inputs vor
    Geometrie_sonder
    Akkumulation_sonder
    Dichte_sonder
    Rest_normal
    % Modell
    aktuell=pwd;
    eval([sprintf('cd %s',PfadCouran)])
    ! run a.exe
    eval([sprintf('cd %s',aktuell)])
    status=0;
    h1=toc;
    while status == 0
        [status,message,messageid] =
movefile([PfadCouran,'/resul'],[PfadArchiv,'/resul/resul_',num2str(durchlauf)]);
        h2=toc;
        if h2-h1 > 30
            fprintf('LANGER DURCHLAUF???\t%4.0f\n',durchlauf);
        else
        end
    end

movefile([PfadCouran,'/resul2'],[PfadArchiv,'/resul/resul2_',num2str(durchlauf)]);
    waitbar(durchlauf/Durchlaeufe);
end
close(h);
% x-Gitter
global x
x=min(xGeo):0.1:max(xGeo);
% zur Dokumentation: schreibe x in Datei
aktuell=pwd;
eval([sprintf('cd %s',PfadArchiv)])
fid=fopen('output/x.txt','wt');
for i=1:length(x)
    fprintf(fid,'%3.1f\n',x(i));
end
fclose(fid);
eval([sprintf('cd %s',aktuell)])
% Trajektorien % Isochrone extrahieren & auf Gitter interpolieren
Extraktion2(Durchlaeufe)
global numtra AGE
% Mittelwert und Percentile der Trajektorien
Mean_Perc_Tra(Durchlaeufe)
global Testverteilungen_tra
% Mittelwert und Percentile der Isochronen
Mean_Perc_Iso(Durchlaeufe)
global Testverteilungen_iso
% Umdrehen der Trajektorien (sprich: in einer bestimmten Tiefe am FL-Ende:
% Woher kann das Eis überall kommen)
Umdrehen
global numdepth zGrid xx
% Datierung bearbeiten
Datierung
% Datierung nach couran
Datierung_couran(Durchlaeufe)
toc

function Input_Bootstrap(Input)
% In der Datei Input.txt muessen folgende Pfade angegeben sein
% Geometrie-Input:
% x | Oberflaeche | dOberflaeche | Eisdicke | dEisdicke
% Akkumulations-Input:
% x | Akkrate [mwe/a] | dAkkrate
% Dichte-Input1:
% x1 (immer derselbe Wert) | Tiefe | Dichte | dDichte
% Dichte-Input2:
% analog mit x2
% Pfade:
% Pfad fuer Archiv... Im Archiv muessen folgende Unterordner liegen: \resul \tra
\iso \input \output \umdrehen \datierung
% Pfad fuer couran (Wo liegt das Programm)
% restlicher Input:
% Divergenz | dDivergenz
% q0 | dq0
% charAkk | charAkk2
% EndQuell | 0
% charAkk=1: Akkrate steigt monoton
% charAkk=-1: Akkrate faellt monoton
% charAkk=0: Akkrate ist frei im zufaelligen Verlauf
% charAkk2 ist nur drin, damit das Programm nicht meckert
% EndQuell: Ort, an dem die Trajektorien den Tiefenintervallen zugeordnet
%           werden sollen, um die Quellregion zu bestimmen
global xGeo OFGeo dOFGeo HGeo dHGeo xAkk bAkk dbAkk xDens1 zDens1 Dichte1 dDichte1
xDens2 zDens2 Dichte2 dDichte2 Divergenz dDivergenz q0 dq0 charAkk PfadArchiv
PfadCouran EndQuell
fid_Input=fopen(Input,'r');
GeometrieInput=fgetl(fid_Input);
AkkumulationInput=fgetl(fid_Input);
DichteInput1=fgetl(fid_Input);
DichteInput2=fgetl(fid_Input);
PfadeInput=fgetl(fid_Input);
RestInput=fgetl(fid_Input);
fclose(fid_Input);
inGeometrie=load(GeometrieInput);
inAkkumulation=load(AkkumulationInput);
inDichte1=load(DichteInput1);
inDichte2=load(DichteInput2);
inRest=load(RestInput);
xGeo=inGeometrie(:,1); OFGeo=inGeometrie(:,2); dOFGeo=inGeometrie(:,3);
HGeo=inGeometrie(:,4); dHGeo=inGeometrie(:,5); clear inGeometrie GeometrieInput
xAkk=inAkkumulation(:,1); bAkk=inAkkumulation(:,2); dbAkk=inAkkumulation(:,3);
clear inAkkumulation AkkumlationsInput
xDens1=inDichte1(:,1); zDens1=inDichte1(:,2); Dichte1=inDichte1(:,3);
dDichte1=inDichte1(:,4); clear inDichte1 DichteInput1
xDens2=inDichte2(:,1); zDens2=inDichte2(:,2); Dichte2=inDichte2(:,3);
dDichte2=inDichte2(:,4); clear inDichte2 DichteInput2
Divergenz=inRest(1,1); dDivergenz=inRest(1,2); q0=inRest(2,1); dq0=inRest(2,2);
charAkk=inRest(3,1); EndQuell=inRest(4,1); clear inRest RestInput
fid_Pfade=fopen(PfadeInput,'r');
PfadArchiv=fgetl(fid_Pfade); PfadCouran=fgetl(fid_Pfade);
fclose(fid_Pfade); clear PfadeInput

function Geometrie_sonder
% Oberflaeche:
% nimm die Werte aus dem Input; die Variation wird dann folgendermaßen
% bestimmt:
% Innerhalb des Fehlers an Start- und Endpunkt werden (normalverteilt) Offsets
generiert.
% Dann wird eine Gerade durch diese beiden Punkte gelegt und auf die Hoehen
% addiert

global xGeo OFGeo dOFGeo HGeo dHGeo durchlauf PfadArchiv PfadCouran
Offset1=dOFGeo(1)*randn; Offset2=dOFGeo(length(dOFGeo))*randn;
% Gerade:
for i=1:length(xGeo)
    y=(Offset2-Offset1)/(xGeo(length(xGeo))-xGeo(1))*(xGeo(i)-xGeo(1))+Offset1;
    OF(i,1)=OFGeo(i)+y; clear y
end
% Das Bett wird folgendermaßen variiert:
% 1. Offset
% 2. eine Gerade
Offset=0.5*mean(dHGeo)*randn; Offset1=0.5*randn*dHGeo(1);
Offset2=0.5*randn*dHGeo(length(dHGeo));
for i=1:length(xGeo)
    y=(Offset2-Offset1)/(xGeo(length(xGeo))-xGeo(1))*(xGeo(i)-xGeo(1))+Offset1;
    H(i)=HGeo(i)+y+Offset;
end
% Beschreibe das File fuer couran und ein Archiv-file, das erhalten bleibt
fid=fopen([PfadCouran,'/GEOM.txt'],'wt');
eval([sprintf('fid_archiv=fopen([PfadArchiv,''/input/Geom_%d.txt''],''wt'');',durc
hlauf)]);
```

```
for i=1:length(xGeo)
    fprintf(fid,'%3.2f\t%4.2f\t%4.2f\n',[xGeo(i), OF(i)-H(i), OF(i)]);
    fprintf(fid_archiv,'%3.2f\t%4.2f\t%4.2f\n',[xGeo(i), OF(i)-H(i), OF(i)]);
end
fclose(fid); fclose(fid_archiv);

function Akkumulation_sonder
% Generiere zufaelligen Input fuer Akkumulation (ggf. unter Beachtung der
% vorgegebenen Monotonie)
global xAkk bAkk dbAkk durchlauf PfadArchiv PfadCouran charAkk
Offset1=dbAkk(1)*randn; Offset2=dbAkk(length(dbAkk))*randn;
for i=1:length(xAkk)
    y=(Offset2-Offset1)/(xAkk(length(xAkk))-xAkk(1))*(xAkk(i)-xAkk(1))+Offset1;
    b(i)=bAkk(i)+y;
end
% Beschreibe das File fuer couran und ein Archiv-file, das erhalten bleibt
fid=fopen([PfadCouran,'/MASSB.txt'],'wt');
eval([sprintf('fid_archiv=fopen([PfadArchiv,''/input/Massb_%d.txt''],''wt'');',dur
chlauf)]);
for i=1:length(xAkk)
    fprintf(fid,'%3.2f\t%1.3f\n',[xAkk(i),b(i)]);
    fprintf(fid_archiv,'%3.2f\t%1.3f\n',[xAkk(i),b(i)]);
end
fclose(fid); fclose(fid_archiv);

function Dichte_sonder
% Generiere zufaelligen Input fuer Dichte
% Hier aber nicht normalverteilt, sondern die Dichteverteilung verschoben
% in der Tiefe
global xDens1 zDens1 Dichte1 dDichte1 xDens2 zDens2 Dichte2 dDichte2 PfadArchiv
PfadCouran durchlauf
% Idee: verschiebe alle Tiefenwerte um einen normalverteilte
% Zufallsstrecke: (std dieser Verteilung: z.B. 2m)
% an Punkt 1
Offset=2*randn; zD1=zDens1+Offset; D1=Dichte1;
% Drei moegliche Varianten:
% 1. Offset ist negativ ==> Tiefenverteilung faengt bei z < 0 an
if zD1(1) < 0
    % Erweitere unten bis zur maximalen Tiefe:
    while max(zD1) < max(zDens1)
        r=length(zD1); zD1=[zD1; zD1(r)+1];
        if D1(r) > 0.9
            D1=[D1; D1(r)];
        else
            D1_neu=(D1(r-1)-D1(r))/(zD1(r-1)-zD1(r))*1+D1(r);
            D1=[D1; D1_neu];
            clear D1_neu
        end
    end
    % interpoliere wieder linear auf die urspruenglichen Dichte Koordinaten
    D1=interp1(zD1,D1,zDens1);
% Oder: Offset ist positiv:
elseif zD1(1) > 0
    % Erweitere oben bis 0:
    while min(zD1) > 0
        if D1(1) < 0.35
            D1=[D1(1); D1];
        else
            D1_neu=(D1(2)-D1(1))/(zD1(2)-zD1(1))*(-1)+D1(1);
            D1=[D1_neu; D1];
            clear D1_neu
        end
        zD1=[zD1(1)-1; zD1];
    end
    % interpoliere wieder linear auf die urspruenglichen Dichte Koordinaten
    D1=interp1(zD1,D1,zDens1);
% Oder: Offset=0: Dann nimm einfach die urspruengliche Dichteverteilung:
else
    D1=Dichte1;
end
% an Punkt 2
Offset=2*randn; zD2=zDens2+Offset; D2=Dichte2;
if zD2(1) < 0
    % Erweitere unten bis zur maximalen Tiefe:
    while max(zD2) < max(zDens2)
        r=length(zD2); zD2=[zD2; zD2(r)+1];
        if D2(r) > 0.9
            D2=[D2; D2(r)];
        else
            D2_neu=(D2(r-1)-D2(r))/(zD2(r-1)-zD2(r))*1+D2(r);
            D2=[D2; D2_neu];
            clear D2_neu
        end
    end
    % interpoliere wieder linear auf die urspruenglichen Dichte Koordinaten
    D2=interp1(zD2,D2,zDens2);
elseif zD2(1) > 0
    % Erweitere oben bis 0:
    while min(zD2) > 0
        if D2(1) < 0.35
            D2=[D2(1); D2];
        else
            D2_neu=(D2(2)-D2(1))/(zD2(2)-zD2(1))*(-1)+D2(1);
            D2=[D2_neu; D2];
            clear D2_neu
        end
        zD2=[zD2(1)-1; zD2];
    end
    % interpoliere wieder linear auf die urspruenglichen Dichte Koordinaten
    D2=interp1(zD2,D2,zDens2);
else
    D2=Dichte2;
end
% Beschreibe das File fuer couran und ein Archiv-file, das erhalten bleibt
fid=fopen([PfadCouran,'/DENS.txt'],'wt');
eval([sprintf('fid_archiv=fopen([PfadArchiv,''/input/Dens_%d.txt''],''wt'');',durc
hlauf)]);
for i=1:length(xDens1)
    fprintf(fid,'%3.2f\t%3.2f\t%1.3f\n',[xDens1(i), zDens1(i), D1(i)]);
    fprintf(fid_archiv,'%3.2f\t%3.2f\t%1.3f\n',[xDens1(i), zDens1(i), D1(i)]);
end
fprintf(fid,'0\t0\t0\n');
fprintf(fid_archiv,'0\t0\t0\n');
for i=1:length(xDens2)
    fprintf(fid,'%3.2f\t%3.2f\t%1.3f\n',[xDens2(i), zDens2(i), D2(i)]);
    fprintf(fid_archiv,'%3.2f\t%3.2f\t%1.3f\n',[xDens2(i), zDens2(i), D2(i)]);
end
fprintf(fid,'0\t0\t0\n');
fprintf(fid_archiv,'0\t0\t0\n');
fclose(fid); fclose(fid_archiv);

function Rest_normal
% Generiere zufaelligen Input fuer Divergenz & q0
global Divergenz dDivergenz q0 dq0 PfadArchiv PfadCouran durchlauf
test=0;
while test == 0
    div=Divergenz+dDivergenz*randn;
    % Verhindere, dass div zu gross wird
    if (div < 0.8) & (div > 0)
        test=1;
    else
    end
end
test=0;
while test == 0
    qnull=q0+dq0*randn;
    % Verhindere, dass qnull zu klein wird
    if qnull >= 5
        test=1;
    else
    end
end
% Beschreibe erstens response2 neu und zweitens ein Sammel-File fuers
% Archiv:
response2=load([PfadCouran,'\response2']);
response2(4)=div; response2(5)=qnull;
fid=fopen([PfadCouran,'\response2'],'wt');
for i=1:5
    fprintf(fid,'%3.2f\n',response2(i));
end
fclose(fid);
out=load([PfadArchiv,'/input/Div_q0.txt']);
out=[out; div, qnull]; [r,c]=size(out);
fid=fopen([PfadArchiv,'/input/Div_q0.txt'],'wt');
```

```
for i=1:r
    fprintf(fid,'%3.2f\t%3.2f\n',[out(i,1), out(i,2)]);
end
fclose(fid);

function Extraktion2(Durchlaeufe)
% Zerlege die resul-dateien in einzelne Trajektorien/Isochrone
% und interpoliere
% Am Ende liegen die Daten in einzelnen Dateien vor:
% Pro Trajektorie je eine Datei, in der zeilenweise die interpolierten
% Hoehen an den x-Gitter-Werten stehen, also:
% Eine Zeile = Ein Durchlauf
% Eine Spalte = Ein bestimmter x-Wert, alle Durchlaeufe
% analog fuer Isochrone
global PfadArchiv numtra x AGE
aktuell=pwd;
eval([sprintf('cd %s',PfadArchiv)])
% Hier wird das Alter der Isochrone festgelegt, die betrachtet werden
%AGE = 10:5:200; mi=min(AGE); ma=max(AGE);
% AGE = [11 16 19 24 29 37 44 49];
%AGE = [AGE, 60:10:500];
AGE=[49, 60:10:500];
mi=min(AGE); ma=max(AGE);
hhh=waitbar(0,'Trajektorien und Isochrone extrahieren');
for durchlauf=1:Durchlaeufe
    h1=1;
    % Bereite die Vektoren fuer die Isos vor:
    for h=1:length(AGE)
        eval([sprintf('ISO_%d=[];',h)])
    end
    eval([sprintf('in=load(''resul/resul_%d''); ! rm resul/resul_%d',durchlauf,
durchlauf)])
    [r,c]=size(in);
    %Trajektorien rausfischen:
    p=1; k=1; test=in(1,c); zdat=[]; tdat=[];
    for i=1:r
        if test ~= in(i,c)
            % checke, ob Trajektorie bis zum Ende geht
            if (in(i-1,1) < max(x)) & (abs(in(i-1,1)-max(x)) < 3)
                eval([sprintf('x_%d(k)=max(x);
z_%d(k)=z_%d(k-1)+in(i-1,4)/in(i-1,3)*(max(x)-in(i-1,1));
u_%d(k)=in(i-1,3);',p,p,p,p)])
                eval([sprintf('zdat=[zdat; z_%d(k)]; tdat=[tdat;
in(i-1,c-1)+(max(x)-in(i-1,1))/in(i-1,3)];',p)])
            elseif in(i-1,1) == max(x)
                zdat=[zdat; in(i-1,2)]; tdat=[tdat; in(i-1,c-1)];
            else
            end
            test=in(i,c); p=p+1; k=1;
        else
        end
        eval([sprintf('x_%d(k)=in(i,1); z_%d(k)=in(i,2);
u_%d(k)=in(i,3);',p,p,p)])
        k=k+1;
        % Ausserdem die Isochrone durchgehen:
        if in(i,c-1) < mi
            h1=1;
        elseif in(i,c-1) > ma
            h1=1;
        else
            for h=h1:length(AGE)
                if AGE(h) == in(i,c-1)
                    eval([sprintf('ISO_%d=[ISO_%d; in(i,1), in(i,2)];',h,h)])
                    h1=h;
                else
                end
            end
        end
    end
    % schreibe die Datierung in Dateien:
    fidz=fopen('datierung/z.txt','at'); fidt=fopen('datierung/t.txt','at');
    for qq=1:length(zdat)
        fprintf(fidz,'%4.4f\t',zdat(qq));
        fprintf(fidt,'%4.1f\t',tdat(qq));
    end
    fprintf(fidz,'\n'); fprintf(fidt,'\n');
    fclose(fidz); fclose(fidt);
    if durchlauf==1
      numtra=p;
    else
    end
    % Trajektorien interpolieren und abspeichern:
    for i=1:p
        eval([sprintf('x_=x_%d; z_=z_%d; u_=u_%d; clear x_%d z_%d
u_%d',i,i,i,i,i,i)])
        r=length(x_);
        % falls die Trajektorie zunaechst nach links (x < 0)
        % laeuft: vernachlaessige die entsprechende Trajektorie!
        % Stattdessen wird in der entsprechenden Datei eine NaN-Zeile
        % gespeichert
        if min(x_) < 0
            eval([sprintf('fid=fopen(''tra/tra_num%d.txt'',''at'');',i)])
            for j=1:length(x)
                fprintf(fid,'NaN\t');
            end
            fprintf(fid,'\n');
            fclose(fid);
            clear x_ z_ u_
            continue
        else
        end
        % Loesche die Teile raus, die erst nach 'links' gegangen sind
        % (sonst versagt spaeter die interp1-Fkt.)
        if u_(1) <= 0
            for j=1:r
                if x_(j) > x_(1)
                    break
                else
                end
            end
            x_=x_(j:r); z_=z_(j:r);
        else
        end
        raus=[];
        for qq=1:length(x_)-1
            for pp=qq+1:length(x_)
                if x_(qq) == x_(pp)
                    raus=[raus; pp];
                else
                end
            end
        end
        rr=length(raus);
        if rr > 1
            % falls mehr als ein gemeinsamer Punkt vorliegt: entferne
            % Trajektorie aus Speicher
            continue
        elseif rr == 1
            % falls nur ein doppelter x_-Wert auftritt: entferne einen der
            % beiden
            [rrr,ccc]=size(x_);
            if raus == 1
                x_=x_(2:rrr); z_=z_(2:rrr);
            elseif raus == rrr
                x_=x_(1:rrr-1); z_=z_(1:rrr-1);
            else
                x_=[x_(1:raus-1), x_(raus+1:rrr)]; z_=[z_(1:raus-1),
z_(raus+1:rrr)];
            end
        else
            % falls kein doppelter x_-Wert vorliegt: weiter gehts
        end
        r=length(x_);
        % interp1 funktioniert nur mit mindestens 2 Datenpunkten, also
        % ebenfalls NaNs in die Datei
        if r < 2
            eval([sprintf('fid=fopen(''tra/tra_num%d.txt'',''at'');',i)])
            for j=1:length(x)
                fprintf(fid,'NaN\t');
            end
            fprintf(fid,'\n');
            fclose(fid);
            clear x_ z_ u_
            continue
```

```
    else
    end
    z=interp1(x_,z_,x); r=length(x);

    % ggf. wird noch ein Datenpunkt hinten angehaengt: Die Trajektorien
    % werden ja zumeist "zu frueh" abgebrochen
    if (isnan(z(r)) == 1) & (isnan(z(r-30)) ~= 1)
        addpath(aktuell)
        z=Extrapolation_am_Rand(z);
    else
    end

    eval([sprintf('fid=fopen(''tra/tra_num%d.txt'',''at'');',i)])
    for j=1:length(z)
        fprintf(fid,'%4.4f\t',z(j));
    end
    fprintf(fid,'\n');
    fclose(fid);
    clear x_ z_ z u_
end
% Isochrone:
for h=1:length(AGE)
    eval([sprintf('ISO=ISO_%d; clear ISO_%d',h,h)])
    % Auch hier: falls die Isochrone nur einen Eintrag hat:
    % ersetze sie durch NaNs:
    [rI,cI]=size(ISO);
    if rI < 2
        eval([sprintf('fid=fopen(''iso/iso_alter%d.txt'',''at'');',AGE(h))])
        for j=1:length(x)
            fprintf(fid,'NaN\t');
        end
        fprintf(fid,'\n');
        fclose(fid);
        clear ISO rI cI
        continue
    else
    end
    raus=[];
    for qq=1:length(ISO(:,1))-1
        for pp=qq+1:length(ISO(:,1))
            if ISO(qq,1) == ISO(pp,1)
                raus=[raus; pp];
            else
            end
        end
    end
    rr=length(raus);
    if rr > 1
        continue
    elseif rr == 1
        [rrr,ccc]=size(ISO);
        if raus == 1
            ISO=ISO(2:rrr,:); raus=raus-1;
        elseif raus == rrr
            ISO=ISO(1:rrr-1,:);
        else
            ISO=[ISO(1:raus-1,:); ISO(raus+1:rrr,:)];
        end
    else
    end
    % interpoliere
    z=interp1(ISO(:,1),ISO(:,2),x); r=length(z);

    % ggf. wird noch ein Datenpunkt hinten angehaengt: Die Trajektorien
    % werden ja zumeist "zu frueh" abgebrochen
    if (isnan(z(r)) == 1) & (isnan(z(r-30)) ~= 1)
        addpath(aktuell)
        z=Extrapolation_am_Rand(z);
    else
    end

    eval([sprintf('fid=fopen(''iso/iso_alter%d.txt'',''at'');',AGE(h))])
    for j=1:length(x)
        fprintf(fid,'%4.4f\t',z(j));
    end
    fprintf(fid,'\n');
    fclose(fid);
        clear ISO rI cI
    end
    clear x_* z_*
    waitbar(durchlauf/Durchlaeufe);
end
close(hhh);
eval([sprintf('cd %s',aktuell)])

function z = Extrapolation_am_Rand(z)
global x
r=length(z);
% Ab wo genau ist z NaN?
for i=r-40:r
    if isnan(z(i)) == 1
        break
    else
    end
end
i0=i;
for i=i0:r
    z(i)=(z(i0-1)-z(i0-2))/(x(i0-1)-x(i0-2))*(x(i)-x(i0-2))+z(i0-2);
end

function Mean_Perc_Tra(Durchlaeufe)
global numtra PfadArchiv Testverteilungen_tra x
aktuell=pwd;
eval([sprintf('cd %s',PfadArchiv)])
Testverteilungen_tra=0;
fid_Testverteilungen=fopen('Testverteilungen_tra.txt','wt');
h=waitbar(0,'Trajektorien mitteln');
for j=1:numtra
    eval([sprintf('in=load(''tra/tra_num%d.txt'');',j)])
    [r,c]=size(in);

    % 1 Zeile enthält 1 Trajektorie
    % 1 Spalte enthält alle z-Werte zu einem x-Wert
    % Spaltenweise wird nun gemittelt und der Fehler bestimmt
    for p=1:c
        % Fische die NaNs heraus
        Werte=[];
        for k=1:r
            if isnan(in(k,p)) == 0
                Werte=[Werte; in(k,p)];
            else
            end
        end
        [rW,cW]=size(Werte);

        % suche zufaellig ein paar Puntke raus, an denen zu Testzwecken der
        % "Werte"-Vektor abgespeichert wird:
        test=randn;
        if (test >= 3) & (rW ~= 0)
            Testverteilungen_tra=Testverteilungen_tra+1;

eval([sprintf('fid=fopen(''Testverteilung_Tra_%d.txt'',''wt'');',Testverteilungen_
tra)])
            for rr=1:length(Werte)
                fprintf(fid,'%4.4f\n',Werte(rr));
            end
            fclose(fid);
            % Notiz fuer spaetere Verwendung bei der Darstellung:
            fprintf(fid_Testverteilungen,'%4.0f\t%3.2f\n',[j,x(p)]);
        else
        end
        if rW ~= 0
            % Bestimme den Mittelwert
            Mittel=mean(Werte);
            % Percentile:
            % Vorerst wird statt der Percentile die Standardabweichung
            % verwendet:
            dMittel=std(Werte);
        else
            % sonst: NaN
            Mittel=interp1([0 1],[0 1],3);
            dMittel=interp1([0 1],[0 1],3);
        end
        Z(p)=Mittel; dZ(p)=dMittel;
```

```
        clear Werte Mittel dMittel
    end
    eval([sprintf('fid=fopen(''output/tra_%d_Mittel.txt'',''wt'');',j)])
    for i=1:length(Z)
        fprintf(fid,'%4.2f\t%4.2f\n',[Z(i), dZ(i)]);
    end
    fclose(fid);
    clear in Z dZ
    waitbar(j/numtra);
end
close(h);
fclose(fid_Testverteilungen);
eval([sprintf('cd %s',aktuell)])

function Mean_Perc_Iso(Durchlaeufe)
global numtra PfadArchiv AGE Testverteilungen_iso x
aktuell=pwd;
eval([sprintf('cd %s',PfadArchiv)])
Testverteilungen_iso=0;
fid_Testverteilungen=fopen('Testverteilungen_iso.txt','wt');
h=waitbar(0,'Isochrone Mitteln');
for j=1:length(AGE)
    eval([sprintf('in=load(''iso/iso_Alter%d.txt''); ! rm
iso/iso_Alter%d.txt',AGE(j),AGE(j))])
    [r,c]=size(in);
    % 1 Spalte enthält 1 Trajektorie
    % 1 Zeile enthält alle z-Werte zu einem x-Wert
    % Spaltenweise wird nun gemittelt und der Fehler bestimmt
    for p=1:c
        % Fische die NaNs heraus
        Werte=[];
        for k=1:r
            if isnan(in(k,p)) == 0
                Werte=[Werte; in(k,p)];
            else
            end
        end
        [rW,cW]=size(Werte);

        % suche zufaellig ein paar Puntke raus, an denen zu Testzwecken der
        % "Werte"-Vektor abgespeichert wird:
        test=randn;
        if (test >= 3) & (rW ~= 0)
            Testverteilungen_iso=Testverteilungen_iso+1;

eval([sprintf('fid=fopen(''Testverteilung_Iso_%d.txt'',''wt'');',Testverteilungen_
iso)])
            for rr=1:length(Werte)
                fprintf(fid,'%4.4f\n',Werte(rr));
            end
            fclose(fid);
            % Notiz fuer spaetere Verwendung bei der Darstellung:
            fprintf(fid_Testverteilungen,'%4.0f\t%3.2f\n',[AGE(j),x(p)]);
        else
        end
        if rW ~= 0
            % Bestimme den Mittelwert
            Mittel=mean(Werte);
            % Percentile:
            % Vorerst wird statt der Percentile die Standardabweichung
            % verwendet:
            dMittel=std(Werte);
        else
            % sonst: NaN
            Mittel=interp1([0 1],[0 1],3);
            dMittel=interp1([0 1],[0 1],3);
        end
        Z(p)=Mittel; dZ(p)=dMittel;
        clear Werte Mittel dMittel
    end
    eval([sprintf('fid=fopen(''output/iso_Alter%d_Mittel.txt'',''wt'');',AGE(j))])
    for i=1:length(Z)
        fprintf(fid,'%4.2f\t%4.2f\n',[Z(i), dZ(i)]);
    end
    fclose(fid);
    clear in Z dZ
    waitbar(j/length(AGE));
end
close(h);
fclose(fid_Testverteilungen);
eval([sprintf('cd %s',aktuell)])

function Umdrehen
% Sortiere die Trajektoriendaten dahingehend um, das sozusagen aus einer
% gewissen Tiefe am Ende der FL eine Rueckwaerts-Trajektorie startet, die
% dann oben an der OF einen gewissen x-Fehler aufweist.
global numtra PfadArchiv x xGeo OFGeo HGeo numdepth zGrid EndQuell xx
aktuell=pwd;
eval([sprintf('cd %s',PfadArchiv)])
for i=1:length(x)
    if abs(x(i) - EndQuell) < 0.1
        iQuell=i;
        break
    else
    end
end
xx=x(1:iQuell);
OF=interp1(xGeo,OFGeo,xx); H=interp1(xGeo,HGeo,xx);
% Definiere Tiefen-Intervalle als "Startpunkte":
% hier kann ggf. die Aufloesung verbessert werden
zGrid=fix(OF(length(OF)) - H(length(H))) + 1 : 1 : fix(OF(length(OF))) - 1;
zGrid=flipud(zGrid'); rGrid=length(zGrid); numdepth=zeros(rGrid,1);
h=waitbar(0,'Trajektorien den Tiefenintervallen am FL-Ende zuordnen');
for tra=1:numtra
    eval([sprintf('in=load(''tra/tra_num%d.txt''); ! rm
tra/tra_num%d.txt',tra,tra)])
    [r,c]=size(in);
    % checke zunaechst, ob die Trajektorie hinter EndQuell begonnen hat:
    test=0;
    for i=1:c-1
        if (isnan(in(1,i)) == 1) & (isnan(in(1,i+1)) ~= 1) & (i+1 >= iQuell)
            test=1;
            clear in r c
            break
        else
        end
    end
    if test == 1
        continue
    else
    end
    % Bringe dann die in-Matrix auf die Laenge von xx:
    for i=1:r
        in_tmp(i,:)=interp1(x,in(i,:),xx);
    end
    in=in_tmp; clear in_tmp; [r,c]=size(in);
    for i=1:r
        if isnan(in(i,c)) ~=1
            for j=1:rGrid-1
                if (in(i,c) >= zGrid(j+1)) & (in(i,c) < zGrid(j))
                    numdepth(j)=numdepth(j)+1;

eval([sprintf('fid=fopen(''umdrehen/Intervall%d.txt'',''at'');',j)])
                    if fid ~= -1
                        for k=1:length(xx)
                            fprintf(fid,'%4.4f\t',in(i,k));
                        end
                        fprintf(fid,'\n');
                        fclose(fid);
                    else
                    end
                else
                end
            end
        else
        end
    end
    waitbar(tra/numtra);
end
close(h);
% schau nach, welche Dateien es nun am Ende gibt und setze ggf numdepth=0
for i=1:rGrid-1
    eval([sprintf('fid=fopen(''umdrehen/Intervall%d.txt'',''r'');',i)])
    if fid == -1
```

```
        numdepth(i)=0;
    else
        fclose(fid);
    end
end

% in den Dateien Intervall%d.txt liegen nun die entsprechenden
% Trajektorien, allerdings in Tiefe unter Oberflaeche
% wieder: 1 Zeile = 1 Trajektorie
% als naechstes:
% mittle die entsprechenden Trajektorien:
% Idee: Suche Quell-x, mittle diese, suche Std und suche dann alle die bei
% xQuellmean, xQuellmean+std, xQuellmean-std anfangen (z.B. in einem
% "Kreis" von ca. 2m) und mittle diese dann
h=waitbar(0,'Mittle die umgedrehten Trajektorien');
for j=1:rGrid-1
    if numdepth(j) == 0
        continue
    else
    end
    eval([sprintf('in=load(''umdrehen/Intervall%d.txt''); ! rm
umdrehen/Intervall%d.txt',j,j)])
    [r,c]=size(in);
    % Quell-x suchen
    x0=[]; Index=[];
    for i=1:r
        if isnan(in(i,1)) ~= 1
            x0=[x0; xx(1)]; Index=[Index; 1];
            continue
        else
            for k=2:c
                if (isnan(in(i,k)) ~= 1) & (isnan(in(i,k-1)) == 1)
                    x0=[x0; xx(k)]; Index=[Index; k];
                    break
                else
                end
            end
        end
    end
    % x0 enthaelt nun also das Start-x; Index den zugehoerigen Index
    xM=mean(x0); xL=xM-std(x0); xR=xM+std(x0);
    if xL < min(x); xL=min(x); else; end
    if xR > max(x); xR=max(x); else; end
    if xM < min(x); xM=min(x); else; end
    if xM > max(x); xM=max(x); else; end
    ZL=[]; ZM=[]; ZR=[];
    for i=1:r
        if abs(xx(Index(i)) - xL) <= 2
            ZL=[ZL; in(i,:)];
        else
        end
        if abs(xx(Index(i)) - xM) <= 2
            ZM=[ZM; in(i,:)];
        else
        end
        if abs(xx(Index(i)) - xR) <= 2
            ZR=[ZR; in(i,:)];
        else
        end
    end
   if size(ZL)==[0,0]
       numdepth(j)=0;
       continue
   else
   end
   if size(ZR)==[0,0]
       numdepth(j)=0;
       continue
   else
   end
   if size(ZM)==[0,0]
       numdepth(j)==0;
       continue
   else
   end
    % mittle nun die drei "Trajektorien", damit es ein glatteres Bild gibt:
    for i=1:3
        if i==1; Z=ZL; clear ZL
        elseif i==2; Z=ZM; clear ZM
        elseif i==3; Z=ZR; clear ZR
        else; end
        [r,c]=size(Z);
        % Spalten mitteln
        for p=1:c
            % NaNs raus
            Werte=[];
            for k=1:r
                if isnan(Z(k,p)) ~= 1
                    Werte=[Werte; Z(k,p)];
                else
                end
            end
            [rW,cW]=size(Werte);
            if rW > 0
                Zmean(p)=mean(Werte);
            else
                Zmean(p)=interp1([0 1], [0 1], 3);
            end
        end
        out(:,i)=Zmean'; clear Zmean Z
    end
    eval([sprintf('fid=fopen(''output/Intervall%d_Mittel.txt'',''wt'');',j)])
    for i=1:length(xx)
        fprintf(fid,'%4.4f\t%4.4f\t%4.4f\n',out(i,:));
    end
    fclose(fid);
    clear out in
    waitbar(j/(rGrid-1));
end
close(h);
% zur spaeteren Nachvollziehbarkeit: speichere zGrid in Datei:
fid=fopen('output/!!!Intervalle.txt','wt');
for i=1:length(zGrid)
    fprintf(fid,'%4.1f\n',zGrid(i));
end
fclose(fid);
eval([sprintf('cd %s',aktuell)])

function Datierung
% mittle und plotte die Datierung
global PfadArchiv xGeo HGeo OFGeo
aktuell=pwd;
eval([sprintf('cd %s',PfadArchiv)])
OFEnde=OFGeo(length(OFGeo)); HEnde=HGeo(length(HGeo));
maxTiefe=[]; p=0;
fidz=fopen('datierung/z.txt','r'); fidt=fopen('datierung/t.txt','r');
while (feof(fidz) ~= 1) & (feof(fidt) ~= 1)
    p=p+1;
    z=str2num(fgetl(fidz)); z=OFEnde-z; ztmp=[];
    t=str2num(fgetl(fidt)); ttmp=[];
    for i=1:length(z);
        if (z(i) >= 0) & (z(i) <= HEnde)
            ztmp=[ztmp; z(i)]; ttmp=[ttmp; t(i)];
        else
        end
    end
    maxTiefe=[maxTiefe; max(ztmp)];
    eval([sprintf('z%d=ztmp; t%d=ttmp;',p,p)])
    clear z t ztmp ttmp
end
fclose(fidt); fclose(fidz); ! rm datierung/*
zGitter=0:0.1:max(maxTiefe); zGitter=zGitter';
% Interpoliere
T=[]; hh=waitbar(0,'Interpoliere Datierung');
for i=1:p
    eval([sprintf('z=z%d; t=t%d;',i,i)])
    if length(z) == 1
        clear z t
        continue
    else
    end
    % abcissae should be distinct
    raus=[];
    for j=1:length(z)-1
```

```
        for h=j+1:length(z)
            if z(j) == z(h)
                raus=[raus; h];
            else
            end
        end
    end
    rr=length(raus);
    if rr > 0
        for j=1:rr
            r=length(z);
            if raus(j) == 1
                z=z(2:r); t=t(2:r); raus=raus-1;
            elseif raus(j) >= r
                z=z(1:r-1); t=t(1:r-1);
            else
                z=[z(1:raus(j)-1); z(raus(j)+1:r)]; t=[t(1:raus(j)-1);
t(raus(j)+1:r)];
                if j < rr
                    for qq=j+1:rr
                        raus(qq)=raus(qq)-1;
                    end
                else
                end
            end
        end
    else
    end
    T2=interp1(z,t,zGitter);
    T=[T,T2];
    clear z t T2
    waitbar(i/p);
end
close(hh);
% Mittle die Alter: in T stehen die Alter fuer eine Tiefe in einer Zeile
[r,c]=size(T);
hh=waitbar(0,'Mittle die Datierung');
for i=1:r
    % fische die NaNs raus:
    Werte=[];
    for j=1:c
        if isnan(T(i,j)) ~= 1
            Werte=[Werte; T(i,j)];
        else
        end
    end
    [rW,cW]=size(Werte);
    if rW > 0
        Tmean(i,1)=mean(Werte);
        dTmean(i,1)=std(Werte);
        Tmax(i,1)=max(Werte);
        Tmin(i,1)=min(Werte);
    else
        Tmean(i,1)=interp1([0 1], [0 1], 3);
        dTmean(i,1)=interp1([0 1], [0 1], 3);
        Tmax(i,1)=interp1([0 1], [0 1], 3);
        Tmin(i,1)=interp1([0 1], [0 1], 3);
    end
    clear Werte cW rW
    waitbar(i/r);
end
close(hh)
% Speichere die Datierung in Datei:
fid=fopen('output/Datierung.txt','wt');
for i=1:length(zGitter)
    fprintf(fid,'%3.1f\t%4.0f\t%4.0f\n',[zGitter(i), Tmean(i), dTmean(i)]);
end
fclose(fid);

function Datierung_couran(Durchlaeufe)
% Datierung aus den resul2-Files
global PfadArchiv xGeo HGeo PfadCouran
aktuell=pwd;
eval([sprintf('cd %s',PfadCouran)])
resp=load('response2'); xdat=resp(3); clear resp
eval([sprintf('cd %s',PfadArchiv)])
H=interp1(xGeo,HGeo,xdat);
zGitter=0:0.1:H; zGitter=zGitter'; T=[];
hh=waitbar(0,'Interpoliere couran-Datierung');
for durchlauf=1:Durchlaeufe
    eval([sprintf('in=load(''resul/resul2_%d''); ! rm
resul/resul2_%d',durchlauf,durchlauf)])
    [r,c]=size(in);
    if r < 2
        clear r c in
        continue
    else
    end
    z=in(:,2); t=in(:,3); clear in
    raus=[];
    for i=1:r-1
        for j=i+1:r
            if z(i) == z(j)
                raus=[raus; j];
            else
            end
        end
    end
    rr=length(raus);
    if rr > 0
        for j=1:rr
            r=length(z);
            if raus(j) == 1
                z=z(2:r); t=t(2:r);
            elseif raus(j) >= r
                z=z(1:r-1); t=t(1:r-1);
            else
                z=[z(1:raus(j)-1); z(raus(j)+1:r)]; t=[t(1:raus(j)-1);
t(raus(j)+1:r)];
            end
        end
    else
    end
    T2=interp1(z,t,zGitter);
    T=[T,T2];
    clear z t T2
    waitbar(durchlauf/Durchlaeufe);
end
close(hh);
% Mittle die Alter: in T stehen die Alter fuer eine Tiefe in einer Zeile
[r,c]=size(T); hh=waitbar(0,'Mittle die Datierung');
for i=1:r
    % fische die NaNs raus:
    Werte=[];
    for j=1:c
        if isnan(T(i,j)) ~= 1
            Werte=[Werte; T(i,j)];
        else
        end
    end
    [rW,cW]=size(Werte);
    if rW > 0
        Tmean(i,1)=mean(Werte);
        dTmean(i,1)=std(Werte);
        Tmax(i,1)=max(Werte);
        Tmin(i,1)=min(Werte);
    else
        Tmean(i,1)=interp1([0 1], [0 1], 3);
        dTmean(i,1)=interp1([0 1], [0 1], 3);
        Tmax(i,1)=interp1([0 1], [0 1], 3);
        Tmin(i,1)=interp1([0 1], [0 1], 3);
    end
    clear Werte cW rW
    waitbar(i/r);
end
close(hh)
% Speichere die Datierung in Datei:
fid=fopen('output/Datierung_couran.txt','wt');
for i=1:length(zGitter)
    fprintf(fid,'%3.1f\t%4.0f\t%4.0f\n',[zGitter(i), Tmean(i), dTmean(i)]);
end
fclose(fid);
eval([sprintf('cd %s',aktuell)])
```

```
% Konvertierung von TWTs in Tiefen
%
% Veraenderungen zu Version1:
% Dichte hat tiefenabhaengigen Fehler
%
% Input:
% In externer Datei; zeilenweise muss diese folgendes enthalten:
% Pfad für Dichte an Punkt 1
% Pfad für Dichte an Punkt 2
% tr1 (Spur an Punkt 1)
% tr2 (  "   "    "   2)
% tr_min (normalerweise =1)
% tr_max (Ende des Profils)
% Pfad für Horizonte
% Pfad für Koordinaten
% flipit: Wert=0, falls tr_min mit dem erstem Koordinaten-Set in Koord
%         zusammenfällt und tr_max mit dem letzten
%         Wert=1, falls andersherum
% Rumpf für Dateibezeichnungen beim Output
% IceCores: Anzahl der Datierungen, die man eingeben möchte (falls
%           IceCores=0: Kein weiterer Input erforderlich; falls IceCores>0:
%           entsprechend der Anzahl an Datierungen müssen folgende Paare
%           eingegeben werden:
% Pfad für Eiskern-Datierung x
% Spur der Eiskern-Datierung x
%
% WICHTIG:
% tr1 < tr2 !!!
% tr1 und tr2 müssen NICHT den Endpunkten des Profiles entsprechen (Diese
% sind naemlich tr_min & tr_max)
%
% Außerdem WICHTIG:
% unterster Horizont muss das Felsbett sein (und dieses darf nicht
% fehlen) !!!
% Denn zuerst wird anhand der relativen Tiefe - Dichte - Beziehung die
% Felsbetttiefe errechnet. Dann wird die relative Tiefenskala in eine
% absolute Tiefenskala umgerechnet.

function HorzProcNeu_InputExtern_v2(Input)

fid=fopen(Input,'r');
k=1;
while feof(fid) ~= 1
    tline=fgetl(fid);
    if k==1; dens1=tline;
    elseif k==2; dens2=tline;
    elseif k==3; tr1=str2num(tline);
    elseif k==4; tr2=str2num(tline);
    elseif k==5; tr_min=str2num(tline);
    elseif k==6; tr_max=str2num(tline);
    elseif k==7; data=tline;
    elseif k==8; Koord=tline;
    elseif k==9; flipit=str2num(tline);
    elseif k==10; Rumpf=tline;
    elseif k==11; IceCores=str2num(tline); if IceCores==0; break; else; end;
    elseif k==12; date1=tline;
    elseif k==13; tr_date1=str2num(tline); if IceCores==1; break; else; end;
    elseif k==14; date2=tline;
    elseif k==15; tr_date2=str2num(tline); if IceCores==2; break; else; end;
    elseif k==16; date3=tline;
    elseif k==17; tr_date3=str2num(tline);
    else
    end
    k=k+1;
end
fclose(fid);

% Fehler der Zweiwegelaufzeit (durch Picken / Unschaerfe des Wellenpaketes) (in s)
deltaT=5*10^(-9); deltaT_FB=20*10^(-9);

% Fehler der Datierung (in a) im relevanten Bereich (< 100 a)
deltaDat=3;

% Dichte-Input
% Dichte-Files muessen folgendermassen angeordnet sein:
% relative Tiefenkoordinate | Dichte (g/cm³) | FehlerDichte (g/cm³)
D1=load(dens1);
D2=load(dens2);

if D1(:,1) ~= D2(:,1)
    fprintf('TIEFENSKALA NICHT GLEICH!!! ABBRUCH!!!')
    return
else
end
relDepth=D1(:,1);
dD1=D1(:,3); D1=D1(:,2);
dD2=D2(:,3); D2=D2(:,2);

% Einlesen der Horizonte
fid=fopen(data,'r');
k=0;
while feof(fid) ~= 1
    tline=fgetl(fid);
    if tline(1) == '#'
        continue
    elseif tline(1) == '>'
        k=k+1;
        eval([sprintf('TWT_%d=[];',k)])
        eval([sprintf('trace_%d=[];',k)])
        continue
    else
        tline=str2num(tline);
        eval([sprintf('TWT_%d=[TWT_%d; tline(1)];',k,k)])
        eval([sprintf('trace_%d=[trace_%d; tline(2)];',k,k)])
    end
end
fclose(fid);
numhorz=k;

% Horizonte sollen an allen Spuren zwischen tr_min & tr_max Datenpunkte
% haben ==> lineare Interpolation.
trace_all=tr_min:tr_max; trace_all=trace_all';
for i=1:numhorz
    eval([sprintf('TWT=TWT_%d; tracee=trace_%d;',i,i)])
    TWT_tmp=interp1(tracee,TWT,trace_all);
    eval([sprintf('TWT_%d=TWT_tmp; clear trace_%d',i,i)])
    clear TWT tracee TWT_tmp
end

figure
subplot(2,1,1)
hold on; grid on; box on; axis('ij'); xlabel('trace'); ylabel('TWT [ns]');
xlim([tr_min tr_max])
for i=1:numhorz
    if i~=numhorz
        col=[0 0 1];
    else
        col=[0 0 0];
    end
    eval([sprintf('plot(trace_all,TWT_%d*10^6,''Color'',col)',i)])
end

% ____________________________
% Konvertierung in m und mwe
% ____________________________
eval([sprintf('TWT_FB=TWT_%d; clear TWT_%d',numhorz,numhorz)])
WW=waitbar(0,'Moment ... Die Konvertierung laeuft ...');
for h=1:length(trace_all)
    if trace_all(h) <= tr1
        D=D1;
        % Annahme: 1 Spur = 0.5m
        % bei dieser "Interpolation" (Gleichsetzung der Dichte mit der an
        % den beiden Input-Punkten) wird ein maximaler zusätzlicher
Interpolationsfehler
        % von 0.01 g/cm³ bei einer Distanz von 50m erreicht
        zus=min([0.01; 0.01*abs(trace_all(h)-tr1)*0.5/50]);
        for i=1:length(D)
            dD(i,1)=(dD1(i)^2+(0.01*(trace_all(h)-tr1)*0.5/50)^2)^(1/2);
        end
    elseif trace_all(h) >= tr2
        D=D2; zus=min([0.01; 0.01*abs(trace_all(h)-tr2)*0.5/50]);
        for i=1:length(D)
            dD(i,1)=(dD2(i)^2+zus^2)^(1/2);
        end
```

```
    else
        % linear interpolieren:
        D=(D2-D1)/(tr2-tr1)*(trace_all(h)-tr1)+D1;
        % zusaetzlicher Interpolationsfehler: 0.02 genau zwischen den
        % beiden Input-Punkten, sonst linear abfallend zu diesen beiden Punkten:
        % Ansonsten setzt sich der Fehler nach Gauß aus den beiden
        % Input-Fehlern zusammen
        zus=min([abs(tr1-trace_all(h));
abs(tr2-trace_all(h))])/(abs(tr2-tr1)/2)*0.02;
        for i=1:length(D)
            dD(i,1)=( zus^2 + ((trace_all(h)-tr1)/(tr2-tr1)*dD2(i))^2 +
((1-(trace_all(h)-tr1)/(tr2-tr1))*dD1(i))^2)^(1/2);
        end
    end

    % rechne Dichte in Geschwindigkeit um
    % Außerdem: Berechne direkt den Faktor fuer die Felsbett-TraFo durch
Integration von 1/velo
    % Außerdem: Berechne Felsbett-Tiefe in mwe
    % Außerdem: Berechne Fehler des Konvertierungsfaktors
    Faktor=0;
    dFaktor=0;
    hwe(h,1)=0;
    for j=1:length(D)
        velo(j)=299792458*(1/(1+0.854*D(j)));
        dvelo(j)=299792458*(1/(1+0.854*D(j))^2)*0.845*dD(j);
        if j~=1
            Faktor=Faktor+(relDepth(j)-relDepth(j-1))*1/2*(1/velo(j)+1/velo(j-1));
            hwe(h,1)=hwe(h,1)+(relDepth(j)-relDepth(j-1))*1/2*(D(j)+D(j-1));

dFaktor=dFaktor+(dvelo(j)/velo(j)^2+dvelo(j-1)/velo(j-1)^2)/2*(relDepth(j)-relDept
h(j-1));
        else
        end
    end

    Faktor=1/Faktor;
    dFaktor=dFaktor*Faktor^2;
    zFB(h,1)=TWT_FB(h)/2*Faktor;
    % Fehler von zFB: - Fehler von TWT_FB
    %                 - Fehler von Faktor
    dzFB(h,1)=((Faktor*deltaT_FB/2)^2+(TWT_FB(h)*dFaktor/2)^2)^(1/2);
    hwe(h,1)=hwe(h,1)*zFB(h,1);
    F(h,1)=Faktor;
    Depth=relDepth*zFB(h,1);

    % Berechne Dichte-Gradienten (fuer Fehler der we-Tiefen - s. Fehler.docx)
    for i=2:length(D)-1
        gradD(i)=(D(i+1)-D(i-1))/(Depth(i+1)-Depth(i-1));
    end
    gradD(1)=gradD(2); gradD(length(D))=gradD(length(D)-1);
    % Glaetten:
    for i=1:length(D)
        if (i > 20) & (i < length(D)-20)
            i1=i-20; i2=i+20;
        elseif i <= 20
            i1=1; i2=i+20;
        elseif i >= length(D)-20
            i1=i-20; i2=length(D);
        end
        gradDsmooth(i)=mean(gradD(i1:i2));
    end
    % Integriere den Dichte-Gradienten und den Dichte Fehler
    gradint(1)=0; dDint(1)=0;
    for i=2:length(D)

gradint(i)=gradint(i-1)+(gradDsmooth(i)+gradDsmooth(i-1))/2*(Depth(i)-Depth(i-1));
        dDint(i)=dDint(i-1)+(dD(i)+dD(i-1))/2*(Depth(i)-Depth(i-1));
    end

    % Berechne Fehler der Felsbett-Tiefe in mwe:

dhwe(h,1)=(dDint(length(D))^2+((D(length(D))+gradint(length(D)))*dzFB(h,1))^2)^(1/
2);

    for k=1:numhorz-1
        if h==1; eval([sprintf('z_%d=[]; deltaz_%d=[]; zwe_%d=[];
deltazwe_%d=[];',k,k,k,k)]); else; end
        eval([sprintf('TWT=TWT_%d;',k)])
        if isnan(TWT(h))==1; continue; else; end

        TWTint(1)=0; zwe_int(1)=0;
        rezvelomean(1)=0; drezvelomean(1)=0;
        for j=2:length(D)
            % Integriere reziproke Geschwindigkeit

TWTint(j)=TWTint(j-1)+2*(Depth(j)-Depth(j-1))*1/2*(1/velo(j)+1/velo(j-1));
            % außerdem: Integriere Dichte fuer die Angabe der Tiefe in
            % mwe:
            zwe_int(j)=zwe_int(j-1)+(Depth(j)-Depth(j-1))*1/2*(D(j)+D(j-1));
            % außerdem zur Fehlerberechnung: siehe Dokument (im Prinzip ist
            % damit TWTint=rezvelomean)

rezvelomean(j)=rezvelomean(j-1)+(Depth(j)-Depth(j-1))*1/2*(1/velo(j)+1/velo(j-1));

drezvelomean(j)=drezvelomean(j-1)+(dvelo(j)/velo(j)^2+dvelo(j-1)/velo(j-1)^2)/2*(D
epth(j)-Depth(j-1));
            if TWTint(j) > TWT(h)
                % Integriert wird, bis die gegebene TWT überschritten wird,
                % dann wird zwischen den beiden umfassenden Werten die
                % Tiefe und die Tiefe in we interpoliert.
                z=interp1([TWTint(j-1),TWTint(j)],[Depth(j-1),Depth(j)],TWT(h));
                zwe=interp1([Depth(j-1),Depth(j)],[zwe_int(j-1),zwe_int(j)],z);

rezvmean=interp1([Depth(j-1),Depth(j)],[rezvelomean(j-1),rezvelomean(j)],z)/z;

drezvmean=interp1([Depth(j-1),Depth(j)],[drezvelomean(j-1),drezvelomean(j)],z)/z;
                Faktor=1/rezvmean; dFaktor=drezvmean*Faktor^2;

deltaDichte_int=interp1([Depth(j-1),Depth(j)],[dDint(j-1),dDint(j)],z);

                % Fehler
                deltaz=((Faktor*deltaT/2)^2+(dFaktor*TWT(h)/2)^2)^(1/2);
                % Fehler zwe

gradint_hier=interp1([Depth(j-1),Depth(j)],[gradint(j-1),gradint(j)],z);
                D_hier=interp1([Depth(j-1),Depth(j)],[D(j-1),D(j)],z);

deltazwe=((deltaDichte_int)^2+((D_hier+gradint_hier)*deltaz)^2)^(1/2);
                break
            else
            end
        end
        eval([sprintf('z_%d=[z_%d; z]; zwe_%d=[zwe_%d; zwe]; deltaz_%d=[deltaz_%d;
deltaz]; deltazwe_%d=[deltazwe_%d; deltazwe];',k,k,k,k,k,k,k,k)])
        clear TWT trace_tmp z deltaz zwe TWTint zwe_int rezvelomean drezvelomean
rezvmean drezvmean gradint_hier D_hier deltaDichte_int
    end
    clear D Faktor velo dvelo clear dFaktor gradD gradDsmooth gradint dD zus
dD_int
    waitbar(h/length(trace_all));
end
close(WW);

subplot(2,1,2)
hold on; grid on; box on; axis('ij'); xlabel('trace'); ylabel('Tiefe [m]');
plot(trace_all,zFB,'k')
patch([trace_all;flipud(trace_all)], [zFB-dzFB; flipud(zFB+dzFB)], 'k',
'FaceAlpha',0.5,'LineStyle','none')

for k=1:numhorz-1
    eval([sprintf('z=z_%d; deltaz=deltaz_%d;',k,k)])
    patch([trace_all;flipud(trace_all)], [z-deltaz; flipud(z+deltaz)], 'b',
'FaceAlpha',0.5,'LineStyle','none')
    plot(trace_all,z,'Color',[0 0 1])
    clear z deltaz
end
xlim([tr_min tr_max])

% ____________________________________________________
% Koordinaten:
% ____________________________________________________
WW=waitbar(0,'Moment ... Output laeuft ...');
C=load(Koord);
```

```
[rC,cC]=size(C);
if flipit==1
    for i=1:cC
        C(:,i)=flipud(C(:,i));
    end
else
end
PHI=C(:,1);
LAMBDA=C(:,2);
ZWGS=C(:,3);
YCH=C(:,4);
XCH=C(:,5);
ZCH=C(:,6);
% weise jedem Punkt in C eine Trace zu
s(1,1)=0;
for i=2:rC
    s(i,1)=s(i-1,1)+((YCH(i-1)-YCH(i))^2+(XCH(i-1)-XCH(i))^2)^(1/2);
end
TRA(1,:)=interp1([s(1,1),s(rC,1)], [tr_min,tr_max], s');

for i=1:numhorz-1
    eval([sprintf('PHI_%d=interp1(TRA,PHI,trace_all);',i)])
    eval([sprintf('LAMBDA_%d=interp1(TRA,LAMBDA,trace_all);',i)])
    eval([sprintf('ZWGS_%d=interp1(TRA,ZWGS,trace_all);',i)])
    eval([sprintf('YCH_%d=interp1(TRA,YCH,trace_all);',i)])
    eval([sprintf('XCH_%d=interp1(TRA,XCH,trace_all);',i)])
    eval([sprintf('ZCH_%d=interp1(TRA,ZCH,trace_all);',i)])
    waitbar(i/numhorz/2);
end

PHI_FB=interp1(TRA,PHI,trace_all);
LAMBDA_FB=interp1(TRA,LAMBDA,trace_all);
ZWGS_FB=interp1(TRA,ZWGS,trace_all);
YCH_FB=interp1(TRA,YCH,trace_all);
XCH_FB=interp1(TRA,XCH,trace_all);
ZCH_FB=interp1(TRA,ZCH,trace_all);
waitbar(1/2);
% ____________________________________________
% Output Tiefen:
% ____________________________________________

for i=1:numhorz-1
    filename=[Rumpf,'_Horizont_',num2str(i),'.txt'];

    % Output enthält also:
    % TWT | Spur | Tiefe [m] | eTiefe [m] | Tiefe [mwe] | eTiefe [mwe] | YSchweiz
| XSchweiz
    eval([sprintf('OUT=[TWT_%d, trace_all, z_%d, deltaz_%d, zwe_%d, deltazwe_%d,
YCH_%d, XCH_%d];',i,i,i,i,i,i,i)]);
    [r,c]=size(OUT);

    fid=fopen(filename,'wt');
    for j=1:r

fprintf(fid,'%1.4e\t%4.0f\t%3.2f\t%1.2f\t%3.2f\t%3.2f\t%6.2f\t%5.2f\n',OUT(j,:));
    end

    fclose(fid);
    clear OUT data_tmp r c
    waitbar((numhorz+i)/numhorz/2);
end

filename=[Rumpf,'_FB','.txt'];
fid=fopen(filename,'wt');
OUT=[TWT_FB, trace_all, zFB, dzFB, hwe, dhwe, YCH_FB, XCH_FB, F];
fid=fopen(filename,'wt');
[r,c]=size(OUT);
for j=1:r

fprintf(fid,'%1.4e\t%4.0f\t%3.2f\t%3.2f\t%3.2f\t%3.2f\t%6.2f\t%5.2f\t%1.8e\n',OUT(
j,:));
end
fclose(fid);
waitbar(1); close(WW);


% ____________________________________________
```

```
% Altersbestimmung der Horizonte
% ____________________________________________

if IceCores==0
    return
else
end

% Datierungen müssen in der Form
% z [m] | z [mwe] (oder Nullspalte) | Alter [a]
% vorliegen

for k=1:IceCores
    eval([sprintf('filename=date%d;',k)])
    eval([sprintf('tr_dat=tr_date%d;',k)])
    eval([sprintf('dat=load(''%s'');',filename)])
    ii=length(dat);

    % Berechne Gradienten der Datierung:
    for i=2:ii-1
        grad_dat(i)=(dat(i+1,3)-dat(i-1,3))/(dat(i+1,1)-dat(i-1,1));
    end
    grad_dat(1)=grad_dat(2); grad_dat(ii)=grad_dat(ii-1);

    % Glaette den Gradienten
    % VORSICHT!!! ggf. den Mittelungsbereich überarbeiten...
    for i=1:ii
        if (i > 50) & (i < ii-50)
            i1=i-50; i2=i+50;
        elseif i <= 50
            i1=1; i2=i+50;
        elseif i >= ii-50
            i1=i-50; i2=ii;
        end
        grad(i)=mean(grad_dat(i1:i2));
    end
    for i=1:numhorz-1
        eval([sprintf('z=z_%d; deltaz=deltaz_%d; zwe=zwe_%d;',i,i,i)])

        zdat(i,k)=interp1(trace_all,z,tr_dat);
        deltazdat=interp1(trace_all,deltaz,tr_dat);
        zwedatrad(i,k)=interp1(trace_all,zwe,tr_dat);
        Age(i,k)=interp1(dat(:,1),dat(:,3),zdat(i,k));
        gradd=interp1(dat(:,1),grad,zdat(i,k));

        deltaAge(i,k)=(deltaDat^2+(gradd*deltazdat)^2)^(1/2);
        zwedatdat(i,k)=interp1(dat(:,1),dat(:,2),zdat(i,k));

        clear trace_tmp z zwe deltazdat gradd
    end

    clear filename tr_dat dat ii grad_dat grad
end

file_dat=[Rumpf,'_Alter.txt'];
fid=fopen(file_dat,'wt');
for k=1:IceCores
    for i=1:numhorz-1
        % Output enhält
        % #Bohrkern | #Horizont | Tiefe [m] | Tiefe [mwe] nach Radar |
        %           | Tiefe [mwe] nach Datierung | Alter [a] | FehlerAlter [a]
        OUT=[k,i,zdat(i,k), zwedatrad(i,k), zwedatdat(i,k), Age(i,k),
deltaAge(i,k)];
        fprintf(fid,'%1.0f\t%2.0f\t%3.2f\t%3.2f\t%3.2f\t%4.1f\t%4.1f\n',OUT);
        clear OUT
    end
    fprintf(fid,'\n');
end
fclose(fid);
```

Acknowledgements

I would like to express my gratitude to ...

... Olaf Eisen for supervising my work, for advice and for the opportunity to have an insight to research and in the end even to go to an international conference which – as I think – is not to be taken for granted by an undergraduate student.

... Pascal Bohleber for having been my everyday advisor, critic, coach, coffee fellow, motivator and proofreader for one year at the IUP ... As above, I did not take it for granted and I hope, I did not take up too much of his time.

... Dietmar Wagenbach for the possibility to go to his office in need of advice at any time and for opening my eyes to less obvious aspects of the topics treated in this thesis.

... the members of Limpics and the Ice Group at the IUP and the fellows in 420 for helping me in any possible way, giving me insight to their work and – in the case of Limpics members – for carrying out the additional GPR measurements on Colle Gnifetti in 2010.

... Christian Vincent for providing the code of his model and for advice.

... Martin Lüthi and Ralph Böhlert for providing data and results of their studies.

... Christian Hahn for proofreading. I hope, I could help him a bit as well.

... my family and especially my parents for all possible support during my studies.

... Anik for apples, pears, swimming and especially for interest and proofreading. Sorry for the physicist's habit to ignore articles and some other advice!

Erklärung:

Ich versichere, dass ich diese Arbeit selbstständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Heidelberg, den 01. April 2011 .....