Analysis of in vivo recorded bone strains using the command language PV-WAVE

Citation for published version (APA):

Document status and date:
Published: 01/01/1995

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.
• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.

Download date: 31. Aug. 2023
Analysis of \textit{in vivo} Recorded Bone Strains using the Command Language PV-WAVE

M.H.E. van der Beek

(WFW 95.006)

Performed at:

McCaig Centre for Joint Injury and Arthritis Research
Department of Surgery
University of Calgary
3330 Hospital Drive NW
Calgary, Alberta, Canada T2N 4N1.
## Contents

**Introduction** ................................................................. 1

**1. Materials and methods** .................................................. 3

1.1 Animal model ................................................................. 3
1.2 Strain gage implantation ................................................... 3
1.3 Data collection ................................................................. 4

**2. Strain analysis** ............................................................. 5

2.1 Strain processing .............................................................. 5
2.2 Normal strain distribution .................................................. 6
2.3 Strain gradients ................................................................. 7

**3. The command language PV-WAVE** ..................................... 9

3.1 The command language ....................................................... 9
3.2 Data processing ................................................................. 9
3.3 Graphic functions ............................................................... 9

**4. Analysis of normal strain distribution** ................................ 11

4.1 User demands ................................................................. 11
4.2 PV-WAVE procedures ....................................................... 11
4.3 Critical note ................................................................. 15

**5. The circumferential surface strain gradient** ......................... 16

5.1 User demands ................................................................. 16
5.2 PV-WAVE procedures ....................................................... 16
5.3 Critical note ................................................................. 18

**References** ........................................................................ 20

**Appendices** ....................................................................... 22

---

*Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.*
**List of Variables**

- **base**: type = string.
  Determines the origin of the recorded strain data and is the cluster name of all the data derived from these strain data.
- **num**: type = string.
  This variable defines at what exercising speed the strain data were recorded.
- **samplesa, samplesp**: type = integer
  Respective start and stop indices for determining a sample record to be used for the analysis.
- **step**: type = integer
  Step size for selecting specific points from the sample record.
- **n**: type = integer
  Variable for selecting the number of isopleths to be displayed. N has to be dividable by 50. The isopleths with value -n, -n+50,..., 0,..., n-50, +n are used in the animation and displayed in different colours.

**List of External Procedures**

- **Farshct.15**: procedure that changes the existing colour table into a user defined colour table. The procedure centroid only operates when using this colour table for displaying images.
- **Centroid**: procedure for determining the bone’s centroid. When a radiographic scanned bone cross-section is displayed using colour table 15, the cross-sectional centroid can be determined.
- **Traceconta**: procedure that divides the cross-section into 24 equal angle pie-sectors, and determines the coordinate locations of the 24 points on both the bone surfaces that fix the boundaries of the various sectors.
Abstract

To analyze the vast amount of data obtained from experiments with laboratory animals ('White Leghorn' roosters) in an efficient way, procedures were implemented in the command language PV-WAVE. Bone deformations were recorded \textit{in vivo} with rosette strain gages at three sites of the tarsometatarsus (TMT) of the laboratory animal's left leg during exercise on an treadmill. These data had to be analyzed for (a) distribution of the longitudinal normal strain over the cross-section of the bone, and (b) the distribution of the circumferential surface gradient of this particular normal strain.

The implemented software consists of two sub- and two main procedures. Operations are performed in the main procedures on the raw data with data from the subprocedures and the relevant distributions are visually represented. The analysis in the main procedures consists of a dynamic animation in 2D and 3D. In this way it is possible to process the large data sets and to analyze the results in a simple way. These procedures should be seen as a start for the final analyzing tool.

\textit{Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.}
Introduction

Bone tissue is a carefully balanced structure, as it is a compromise between form and mass requirements to achieve enough strength to accommodate the demands of activity, yet retain the advantage of lightness (Rubin, McLeod, and Bain 1990). On the other hand this structure is very dynamic, and an altered strain milieu can stimulate the bone to change its form and mass. This capacity to perceive and respond to small changes in mechanical environment enables the skeleton to withstand the extremes of physical activity, but this can also have a negative effect.

It is generally recognized that bone tissue needs a certain level of activity for tissue homeostasis. Stress deprivation from immobilisation or static loading (Rubin, Lanyon 1987), can result in significant bone atrophy. Thus fracture plates designed to withstand the mechanical loading put on the fracture site, should also allow for some level of strain magnitude to avoid bone atrophy.

In case of total joint replacement, failures can occur as a result of loosening of the prosthesis. As the prosthesis is much stiffer than the host bone, altered strain magnitudes and strain distributions can cause bone resorption at the implant-bone interface (Maloney et al. 1989). These examples clearly point out the clinical relevance of being able to predict what parameters influence the process of bone remodelling and at what sites this process will occur.

Various studies show that bone tissue is extremely susceptible to dynamic strains (Cowin et al. 1984). The strain environment generated by functional activity is believed to generate a mechanical stimulus by which the bone cell population can assess the skeleton’s structural effectiveness. This information can be used to influence bone morphology (Rubin, McLeod, and Bain 1990). Unfortunately, the nature of the mechanical stimulus that makes the bone alter its shape and structure is poorly understood. Many parameters of the strain milieu have been proposed to be the dominant influence for the skeleton to remodel including strain rate, minimum effective strain, strain history, strain energy density, and the strain tensor.

Recently, Gross and colleagues reported another potent parameter for predicting sites of bony adaptation (Gross, McLeod, and Rubin 1994). They concluded that the circumferential surface gradient of longitudinal normal strain, i.e. how the longitudinal normal strain varies in a circumferential direction, could be a parameter for predicting sites of bony adaptation. To understand the physiological relevance of this parameter, a more detailed characterization of the bone’s functional strain milieu must be obtained by recording strains in vivo. For this purpose White Leghorn roosters are strain gaged in vivo and exercised on a treadmill.

As we are interested in the circumferential surface gradient of longitudinal normal strain, the recorded strain data is analyzed for longitudinal normal strain distribution, and distribution of the circumferential surface gradient of longitudinal normal strain. This paper deals with the analysis tools that were created in the command language PV-WAVE.
Introduction

Chapter 1 briefly deals with the experimental setup and methods that were used. The theories and assumptions made for calculating the normal strains and the various distributions are explained and discussed in chapter 2. Chapters 3 to 5 deal, respectively, with the command language PV-WAVE, procedures for calculating the normal strain distribution and procedures for calculating the surface strain gradient distribution. Listings of the various procedures are included in the appendices.
1 Materials and methods

1.1 Animal model

As stated, the aim of this paper is to investigate whether the circumferential surface gradient of longitudinal normal strain is a potent parameter for predicting sites of exercise-related bony adaptation. To understand more of this parameter's physiological relevance, a more detailed characterization of the skeletons functional strain environment should be obtained. This brings about the relevance of an in vivo animal model. White Leghorn roosters were strain gaged in vivo and exercised on a custom treadmill. As we were interested in exercise-related bone adaptation, adult roosters were used because the effects of growth and exercise-related bone remodelling are intertwined in immature roosters. The functional strain environment engendered during these exercises is recorded with rectangular strain rosettes implanted on the tarsometatarsus (TMT) cortex (see fig.1.1).

For several reasons the rooster was chosen as in vivo animal model. For one, avian bone undergoes modelling and remodelling similar to mammals. The avian bone remodelling rate is approximately twice as high as the human bone remodelling rate. Changes in bone morphology can therefore be observed more quickly. Considering the nature of the experiments, maybe the most important reason for choosing this animal model is because former studies have shown that chickens are readily trained and have the ability to improve their aerobic capacity (Loitz, Zernicke 1992).

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{walking_cycle.png}
\caption{Walking cycle of the rooster leg showing the tarsometatarsus (TMT) and surface strain vectors engendered at the anterior site of the TMT sagittal plane (figure taken from Loitz, Zernicke sub).}
\end{figure}

Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.
### 1.2 Strain gage implantation

All surgery was performed on the left tarsometatarsus (TMT) bone. The bone surface strains engendered during exercise in the rooster TMT were recorded with three rectangular rosette strain gages (fig. 1.2). Under anaesthesia the three rosette strain gages are glued to the TMT's periosteal surface at a medial, anterior, and lateral site after the removal of the periosteum (fig.1.3). When the bone deforms as a result of the animal’s gait, equal deformations in the strain gages will cause a change in their electrical resistance. Because each gage element is admitted to a Wheatstone Bridge, this change in electrical resistance is transformed into a corresponding change in output voltage.

**fig.1.2 Three element rectangular strain rosette.**

**fig.1.3 Orientation of the rectangular rosette strain gages with respect to the longitudinal bone axis.**

All three strain rosettes are positioned in the mid-diaphyseal plane. The lack of mid-shaft muscle attachment in the rooster’s TMT, suggests local muscle influences on the strain gages to be small. Furthermore, this part of the bone represent the most uniform cross-section which is of importance for later strain analysis.

At least three rosette gages should be attached to one transverse plane if technically possible, in order to define the strain milieu more precisely. The use of fewer than three rosette gages makes the strain characterization highly dependent upon the chosen sites of attachment which can result in a distortion of the strain characterization (Gross, McLeod, and Rubin 1992).

### 1.3 Data Collection

The three rectangular rosette strain gages used in the experiments have an electrical resistance $R = 120 \pm 0.5 \, \Omega$, gage factor $k = 2.0$, and a length of 10 mm. Each element of the rosette is admitted to one Wheatstone bridge. With the rooster’s instrumented leg unloaded and relaxed, the initial resistances in each bridge are balanced. Because of this configuration, three resistances are internal to the device, the bridge operation is called 1/4 bridge operation.

The gage leadwires are lead subcutaneously to an exit site at the dorsal mid-thoracic spine and soldered to a 25-pin connector. During each trial, nine data channels were recorded (three
gage elements per rosette). The *in vivo* strains are recorded over a 10 second sampling period with a frequency of 200 Hz and amplified while the exercised rooster walks/runs at a pre-set speed on the treadmill.

*Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.*
2 Strain analysis

2.1 Strain processing

The bone surface deformations generated by functional activity are measured with rosette strain gages. From these recorded strains the desired normal strains are derived. For processing the recorded strains, standard formulae provide a way to compute the longitudinal normal strain $E_{yy}$, transverse normal strain $E_{xx}$, and shear strain $E_{xy}$ at any time (t) of the gait cycle (Dally and Riley 1965).

Consider three gages aligned along the axes A, B, and C as shown in fig.2.1, then the normal strains $E_{yy}$, $E_{xx}$, and shear strain $E_{xy}$ can be obtained by solving the equations

\begin{align*}
\epsilon_a &= \epsilon_{xx} \cos^2 \beta_a + \epsilon_{yy} \sin^2 \beta_a + \epsilon_{xy} \sin \beta_a \cos \beta_a \quad (2.1.1) \\
\epsilon_b &= \epsilon_{xx} \cos^2 \beta_b + \epsilon_{yy} \sin^2 \beta_b + \epsilon_{xy} \sin \beta_b \cos \beta_b \quad (2.1.2) \\
\epsilon_c &= \epsilon_{xx} \cos^2 \beta_c + \epsilon_{yy} \sin^2 \beta_c + \epsilon_{xy} \sin \beta_c \cos \beta_c \quad (2.1.3)
\end{align*}

When assuming the rectangular rosette orientation as shown in fig.2.1 with angles $\beta_a = 45^\circ$, $\beta_b = 90^\circ$, and $\beta_c = 135^\circ$ the normal and shear strains can be determined as

\begin{align*}
\epsilon_{yy} &= \epsilon_b \quad (2.1.4) \\
\epsilon_{xx} &= \epsilon_a + \epsilon_c - \epsilon_b \quad (2.1.5) \\
\epsilon_{xy} &= \epsilon_c - \epsilon_a \quad (2.1.6)
\end{align*}
2.2 Normal strain distribution

The calculated normal strains and shear strains at any time (t) of the gait time are only relevant for the small part of the bone where the strain gage is attached to. Longitudinal normal strain distribution \( \varepsilon_{yy}(x,y,t) \) across the mid-diaphyseal plane is determined by using linear beam theory. This theory provides a way of linearly extrapolating the strain data over a given plane. The resulting strain distribution can in reality only exist in a perfectly homogeneous material (uniform material properties).

Linear beam theory assumes that the TMT shaft can be modelled as a prismatic beam of irregular cross-section. It also assumes that the deformation of the beam is such that plane sections remain plane during deformation and out-of-plane bending can be neglected. As a result of these last assumptions, only axial loads and bending moments can act on the bone. Linear beam theory does not account for shear strains induced by mechanical loads in the transverse direction. This is a severe limitation when using these strain data for stress analysis. Only when shear strains do not contribute significantly to the strain environment, linear beam theory can be used safely for strain analysis.

Linear beam theory describes the bone deformation in terms of three parameters. Defining a global coordinate system at the bone's cross-sectional centroid allows calculating the normal strain \( \varepsilon(x,y,t) \) at any point \( (x,y) \) in the cross-section using the expression

---

Note: hence in this paper the distribution of longitudinal normal strain will be referred to as normal strain distribution.
\[ \varepsilon(x, y, t) = \varepsilon_0(x, y, t) + \kappa_x(x, y, t)x + \kappa_y(x, y, t)y \]  

(2.2.1)

where:

- \( \varepsilon_0(x, y, t) \) = average axial strain
- \( \kappa_x \) = radius of curvature about the x-axis
- \( \kappa_y \) = radius of curvature about the y-axis

Basically, \( \varepsilon_0(x, y, t) \) accounts for shifting of the isopleths (= curves with a constant value for strain) with respect to the cross-section's centroid. \( \kappa_x \) and \( \kappa_y \) can rotate the isopleths respectively about the x-axis and y-axis. The three unknown parameters \( \varepsilon_0(x, y, t), \kappa_x(y, t), \) and \( \kappa_y(x, t) \) are determined solving the equation (2.2.1) using the derived normal strains and their respective strain gage coordinate locations.

An alternative method for calculating the strain distribution across the mid-diaphyseal plane would be to use finite element analysis (FEA). Finite element modelling could define the strain environment engendered in the bone more accurately. Comparing strain distributions calculated with FEA and linear beam theory only have shown a deviation of 5% in the case of linear beam theory (Gross 1994). This linear beam approximation of the strain distribution is believed to be an acceptable one.

The advantage of using linear beam theory over finite element analysis is that FEA is based on the assumption that the mechanical behaviour of all the elements, in which the bone is subdivided, is known. Furthermore, boundary conditions such as the continuity of strain should be defined in order to calculate the bone's response to mechanical load. Linear beam theory, however, uses no mechanical properties (Young's modulus, Poisson's ratio) or

---

*Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.*
boundary conditions. Linear beam theory is therefore a more simple and quicker way for calculating the strain distribution.

2.3 Strain gradients

The most important part of the strain data analysis is to determine the distribution of the circumferential surface gradient of longitudinal normal strain. This parameter of the strain milieu could possibly predict sites of bony adaptation. To determine the distribution of this parameter, we simply have to know how the longitudinal normal strain varies in a circumferential direction along the bone’s surface.

After applying linear beam theory, a value for the normal strain $E_{yy}(x,y,t)$ can be appointed to every point in the mid-diaphyseal plane. Both the inner (endosteal) and outer (periosteal) surface are then subdivided into 24 equal-angle pie sectors as is shown in fig.2.3. Both the inner and outer surface have 24 points in common with the boundary-lines of the 24 sectors. Values for the normal strain are calculated with linear beam theory in each of these points.

The normal strain values in every two points that make up the boundary of one sector are used to derive the surface strain gradient in that particular sector using the equation

$$
\varepsilon_{gr}(i) = \frac{(\varepsilon(i+1) - \varepsilon(i))}{d}
$$

where

- $i = 0,1,...,23$
- $d =$ distance along the surface between two successive normal strain values in [mm].

The value for $E_{gr}$ is assigned to the point on the bone surface in the middle of each sector.

Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.
3 The command language PV-WAVE

3.1 The command language

PV-WAVE is a command language that emphasizes visual representation. It has an expression syntax that resembles FORTRAN or C. Like in these command languages, operators are evaluated according to precedence and left to right sequence. Because of its great resemblance to these command languages, PV-WAVE can execute and exchange data with external programs written in FORTRAN and C. These external programs can also call PV-WAVE to perform graphics, data manipulation, and other functions.

3.2 Data processing

Although PV-WAVE resembles FORTRAN and C very much, an important advantage of PV-WAVE over these languages is the fact that the user does not have to declare variables. The variable is automatically declared as a specific data type, as soon as it is assigned a value.

An important feature of PV-WAVE is it being an array-oriented command language. It handles an array as a single entity, rather than all separate numbers. As a result mathematical operations on arrays can be performed in the same manner as on individual elements. This makes it a very convenient command language for analyzing large amounts of data and/or large data sets.

The speed of PV-WAVE is comparable to that of optimized FORTRAN as far as array-operations are considered. However, when scalar operations are used when array operations are appropriate, the penalty is an efficiency degrade by a factor of 30 or more. It is therefore very important to use array-operations rather than loops whenever possible.

3.3 Graphic functions

Many elements within PV-WAVE make this command language an excellent tool for analyzing data by visual representation. Again this is an advantage considering the fact that we are dealing with large data sets.
Some of the graphic functions are:
° plotting
  2D
  3D
  4D
° surface plots
° contour plots
° vector fields
° bar plots
° dynamic animations
° image
  display
  manipulation
  processing
4 Analysis of normal strain distribution

4.1 User demands

Analyzing the recorded strain data for normal strain distribution is not only a logical step in the analyzing process of the surface strain gradients of normal strain, it also provides the user with other useful information. Some other information of interest is: (a) the orientation of the isopleths, (b) the distribution of compressive and tensile strains, and (c) the global strain magnitudes. The surface strain gradients reach extreme values at sites where the normal strain isopleths are oriented perpendicular to the bone surface. This orientation and how it changes in time provides information on where to expect areas with extreme values for $\varepsilon_g$ at a certain sample point in the gait cycle. Information on the global strain magnitudes will enable the user to determine whether the selected exercising speed engenders normal strains of sufficient magnitude to produce noticeable bone adaptation. Finally, the analysis can also be used as overall check by comparing the calculated normal strain distribution with former studies, for example (Loitz, Zernicke sub). To analyze the normal strain distribution properly, the analyzing tool has to meet certain user demands:

- determining the strain gage coordinate locations within the bone cross-section,
- displaying the TMT cross-section with neutral axis and some isopleths to visualize areas of tension and compression,
- the possibility for the user to select a specific range of sample points of the gait cycle for analysis,
- choosing a variable number of isopleths to be displayed,
- the user must, at any time, be able to tell what phase of the animal’s walking cycle corresponds to the analyzed normal strain distribution,
- user-interaction with the animation process.

4.2 PV-WAVE procedures

For meeting these demands the procedures Coord and Animation were written. These two procedures are briefly discussed in the following section. Listings are included in appendix A.

SubProcedure: Coord

Characteristic commands: CURSOR, DC_READ_TIFF, DC_WRITE_TIFF, PLOT, REBIN, TV, TVRD.
Variables: base

Strain gage coordinate locations were determined by a radiographic scanned image of the bone cross-section imported into the procedure. In this image the coordinate locations have already been marked. The image is placed into a coordinate system such that the bone centroid and origin of the coordinate system coincide (fig.4.1). This, however, is not a necessity. The units in the coordinate system can be chosen arbitrarily. As these coordinates are used for determining the parameters \( \varepsilon_0, \kappa, \) and \( \kappa_y \) (linear beam theory), the choice of coordinate system will influence the values for these parameters but not the actual distribution of the normal strain. The command CURSOR (see app. A1) enables the user to determine the strain gage locations interactively by clicking the left mouse button at the marked sites.

![Fig. 4.1 Image of the TMT cross-section displayed in an arbitrary coordinate system.](image)

Main Procedure: Animation

Characteristic commands: DC_READ_TIFF, MOVIE, PLOT, OPLOT, REVERSE, TV, TVRD.

Variables: base, num, samplesa, samplesp, step, n

This procedure enables the user to analyze the normal strain distribution via a dynamic animation. First, the previously determined strain gage coordinate locations and derived normal strain data are imported. An already existing program takes care of calculating the normal and shear strains from the recorded strain data. These strain gage coordinate locations and their respective longitudinal normal strain data are then used to solve for the three unknown parameters, \( \varepsilon_0, \kappa, \) and \( \kappa_y \). This is done for each, by the user selected, sample point of the gait cycle. The user immediately fixes the number of isopleths to be displayed in the animation, by assigning a certain value to variable 'n' in the call for procedure Animation.
The normal strain distribution is only calculated for these values of the normal strain \( \varepsilon_{xy}(x,y,t) \). The number of isopleths calculated for both tension and compression equals \( n_1 = n/50 \) plus the neutral axis (see page A3).

Various elements in the program help the user to determine where to place the normal strain distribution, calculated at a specific sample point, in the walking cycle of the rooster. The display window, see fig. 4.2, shows a sample count (lower left corner) and a plot of the normal strain vs. time at a certain gage (right). Along this plotted curve a small square marks the position in the walking cycle of the rooster that corresponds to the displayed normal strain distribution (left). In this way the user can discriminate in the analysis of the strain distribution between important and less important phases of the rooster’s walking cycle.

After the calculation of the selected isopleths, the following post processing can be divided into 4 sections (see A4):
- image frames for displaying the normal strain distribution at the selected sample points are put together and stored in the variable 'animationA',
- image frames for displaying the 'strain vs. time' plot are composed and stored into variable 'animationB',
- corresponding sets of previously calculated frames are positioned into a larger window. These new image frames are stored into variable 'animationAB',
- using the command MOVIE the actual animation takes place in.

The user-interaction with the animation consists of freezing the animation, restarting the animation at the starting sample point or aborting the animation.

fig. 4.2  Picture taken from the device screen showing an example display window in the analysis of the normal strain distribution. The window shows the user-selected number of isopleths superimposed on the TMT cross-section. Lightly coloured isopleths represent tensile strains separated by the thicker neutral axis from the compressive strains (darkly coloured). The sample count (lower left corner) and the 'strain vs. time' plot connect the normal strain distribution to the corresponding phase of the animal's walking cycle.
4.3 Critical note

Although these procedures provide sufficient possibilities for analyzing the normal strain distribution, there is always room for future improvements. The example window of procedure Animation is made up of the normal strain distribution and a 'strain vs. time' plot. It is not possible in the command language PV-WAVE to run two different animations in separate windows at the same time. The frames for these two parts are calculated separately and then positioned into a third, larger window. This post processing (frame calculations, positioning, and finally the actual animation) takes up quite some time. The procedure Animation could be improved in this area to reduce the necessary processing time.
5 The circumferential surface strain gradient.

5.1 User demands

Because we were interested in the circumferential surface gradient of longitudinal normal strain as a possible parameter for predicting sites of bony adaptation, the following chapter was an important part of the strain analyzing process. In order to gain more insight into the possible relevance of this parameter, we wanted to compare the distribution of the circumferential surface strain gradient with experimental results of bony adaptation. Thus information is necessary on (a) the distribution of the gradient over the surfaces of the analyzed cross-section and (b) the global magnitude of the parameter.

Similar to the analysis of the normal strain distribution, user demands can be defined that have to be met by the analyzing tool:

- determining the strain gage coordinate locations within the bone cross-section,
- displaying the distribution and global magnitude of the surface strain gradients in 3D,
- the possibility to focus on a specific range of sample points of the gait cycle,
- the user must at any time be able to determine what phase of the animal's walking cycle corresponds to the analyzed distribution of the surface strain gradient and
- user-interaction with the animation process.

5.2 PV-WAVE procedures

To analyze the strain data for the distribution of the surface strain gradient, the procedures Coord and Grad are written in the command language PV-WAVE. These procedures are discussed briefly in the following section. The listings of these procedures are included in appendix B.

SubProcedure: Coord

Characteristic commands: CURSOR, DC_READ_TIFF, PLOT, OПLOT, REBIN, TV, TVRD.

Existing procedures used: Farshct.15, centroid

variables: base
The circumferential surface strain gradient

This procedure is very similar to procedure Coord used in the analysis of the normal strain distribution. For the sake of simplicity we will call this procedure Coord*. Coord* will also enable the user to determine the coordinate locations of the strain gages. Again an image of the bone's cross-section, with marked sites for the strain gages, is displayed and via the command CURSOR the user can interactively determine the coordinate locations. However, unlike the former procedure Coord the image is not placed into a user-defined coordinate system. Instead, the already existing device coordinate system (= pixels which make up the display window) is used. Except for this difference, both procedures work in identical ways.

At first, the use of different coordinate systems in otherwise identical procedures seems a bit inconsequent and illogical. However, several reasons can be brought forward to justify this method of work:

1. in the analysis of the normal strain distribution information on the orientation of the isopleths is, among others, of interest. By plotting the coordinate axes as shown in fig.4.1, it is very easy to determine the orientation of the isopleths (see also fig.4.2). A quantitative comparison with respect to isopleth orientation between different phases in the walking cycle of one rooster and between identical phases of different roosters can be made. When the device coordinate system is used instead, it is not possible to plot axes that correspond to this device system. Without the axes only a qualitative comparison between the orientation of isopleths is possible.

2. the already existing procedure Traceconta determines the 24 points of interest on both the endosteal (e, e,) and the periosteal (p, p,) surface in device coordinates. The user is, thus, forced to determine the strain gage coordinate locations in device coordinates since these locations influence the parameters ε, κ, and λ, that are used in linear beam theory. These parameters and the coordinate locations of the 24 determined points per surface are the input for linear beam theory to calculate the normal strain magnitudes in these points.

Main Procedure: Grad

Characteristic commands: DC_READ_TIFF, MOVIE, PLOTS, T3D, TV, TVRD.

Existing procedures used: Traceconta

variables: base, num, samplesa, samplesp, step

This procedure enables the user to analyze how the longitudinal normal strain varies in a circumferential direction along both the inner (endosteal) and outer (periosteal) bone surface. Identically to procedure Animation various data are imported necessary to solve for the three unknown parameters used in linear beam theory. Again, this is done for every, by the user selected, sample point of the gait cycle. The strain gradients are calculated according to equation (2.3.1). However, a linear approximation d* for the distance 'd' along the surface is used. Using simple geometry d* is defined as

Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.
The circumferential surface strain gradient

\[ d'(i) = \sqrt{((x(i+1) - x(i))^2 + (y(i+1) - y(i))^2} \]  \hspace{1cm} (5.2.1)

where:

- \((x(i+1), y(i+1), (x(i), y(i))\) are the respective coordinate locations of the \((i+1)\textsuperscript{th}\) and \(i\textsuperscript{th}\) boundary point of sector \(i\).

- \(d'(i)\) is the linearly approximated length along the surface of sector \(i\).

\(i=0, 1, ..., 23\)

The calculated strain gradients for the endosteal and for the periosteal surface, are appointed to the point which is situated at \(d'/2\), per sector.

In visualizing the strain gradients, the procedure works in a similar way as procedure Animation. Fig.5.1 shows an exemplar window for visualizing the surface strain gradients in 3D:

- \(\circ\) similar as in procedure Animation the user can make a connection with/have a survey of the walking cycle of the rooster and the strain gradients distribution at a specific point in time.

- \(\circ\) a projection of both the bone surfaces in the x-y plane is displayed

- \(\circ\) the calculated strain gradients are connected with their respective point of projection via a coloured vertical line. The legend in the upper right corner will tell what colour corresponds to what range of gradient magnitude, in microstrain \(10^6 \mu \text{E}\).

\textbf{fig.5.1} Picture taken from the device screen showing the display window of procedure Grad. Displayed are the surface strain gradients in 3D which are connected via vertical lines with a projection in the \(z = 0\) plane. The legend in the upper left corner connects the colour of each vertical line to a specific range in strain gradient magnitude.

\textit{Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.}
5.3 Critical note

The generally identical structure of procedures Animation and Grad causes identical problems in both procedures with respect to the post-processing time. The specific way of post-processing, as discussed in paragraph 4.3, will again result in reasonably big process times for an analysis involving ± 20 - 40 sample points.

The magnitude of the strain gradients can only be determined within a pre-defined range. In the example of fig.5.1 this range is 20 microstrain (με). In this current version of procedure Grad the user has to determine the approximate upper and lower limit of the gradient magnitude in a preceding analysis of the strain gradient data, after which he subdivides the intermediate gradient magnitudes into a desired number of ranges. This method of work is rather laborious and could use some improvements. These proposed improvements could not be implemented in the available time in this version of the procedure.
References


Gross T.S., Maccaig Centre for Joint Injury and Arthritis Research, Calgary (Canada) 1994. Private communication


---

*Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.*
Samenvatting

Om de grote hoeveelheid data verkregen uit experimenten aan proefdieren op een efficiënte manier te kunnen analyseren, werden in de computer taal PV-WAVE enkele procedures geïmplementeerd. Tijdens oefeningen op een lopende band werden op drie plaatsen op de linker poot van het proefdier ('White Leghorn' hanen) *in vivo* botdefonnaties gemeten m.b.v. rekstrookjes. Deze data moet geanalyseerd worden op (a) de verdeling van de axiale normaal rek over het dwarsoppervlak van het bot en (b) de verdeling van de gradient van deze normaal rek in de omtreksrichting van het bot.

De geïmplementeerde software bestaat uit twee sub- en twee hoofdprocedures. In de hoofdprocedures worden o.a. m.b.v. gegevens uit de subprocedures bewerkingen op de ruwe data uitgevoerd en worden de relevante verdelingen visueel geregpresenteerd. De analyse in de hoofdprocedures bestaat uit een dynamische animatie van de verdelingen in 2D en 3D. Op deze manier is het mogelijk geworden om de omvangrijke data sets te verwerken en de gegevens op een eenvoudige manier te analyseren. Deze procedures moeten gezien worden als een eerste aanzet tot het uiteindelijke analyse tool.

*Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.*
Appendices

Longitudinal Normal Strain Distribution .............................. A1
  subprocedure Coord.pro ............................................. A1
  main procedure Animation.pro ................................. A2

Circumferential surface strain gradient ............................. B1
  subprocedure Coord.pro ............................................. B1
  main procedure Grad.pro ......................................... B2

Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.
pro coord,base

coord="/homes/stefan/pvwave/maurice/data"+base+"/coord.dat"

image=bytearr(300,300)
new_image=bytearr(600,600)

; Program for interactively determining the coordinates of the strain gages. The coordinates are then saved in file "coord.dat"

loadct,5
window,0./pixmap,xsize=250,ysize=250

status=DC_READ_TIFF('/homes/stefan/pvwave/maurice/image'+base+"/"+base+'_mini.tif', mini)
TV, (mini)
tmt=TVRD(0,0,235,200)

DELETE,0
window,1./pixmap,xsize=300,ysize=300
plot,findgen(2), /nodata,xrange=[-6,6],yrange=[-6,6]
TV,tmt,33,41/device
axis,-6,0,yaxis=0
axis,0,6,xaxis=1
axis,0,-6,xaxis=0
image=TVRD(0,0,300,300)
status=DC_WRITE_TIFF('/homes/stefan/pvwave/maurice/image'+base+"/"+base+'_plot.tif', image)

DELETE,1
window,2,xsize=600,ysize=600,$
   xpos=100,ypos=100,title="point coordinates..."
new_image=REBIN(image,600,600)
TV,new_image

; interactively reading the coordinates of the rosettes by clicking the markings on the bone surface in the order: M,A,L

cursor,x1,y1,3
print,'medial: ',x1,y1
cursor,x2,y2,3
print,'anterior: ',x2,y2
cursor,x3,y3,3
print,'lateral: ',x3,y3

DELETE,2
openw,1,coord
printf,1,x1,y1,x2,y2,x3,y3
close,1

end

Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.
This program reads the normal strains calculated by "strain.pro" and makes a 2D animation of the normal strain distribution. Variables to enter:

- 'base' - determines origin of data,
- 'samplesa', 'samplesp' - respective start and stop indices for determining a sample trajectory for the animation,
- 'step' - determines how many steps to take, starting at 'samplesa' and ending at 'samplesp', for the animation,
- 'n' - maximum isopleth to be displayed,
- 'n', 0, ..., +n are the isopleths to be displayed with steps of 50 microstrain.

```
N1=n/50
Nstep=N/step
x=(findgen(121)/10-6)
eyy1=fltarr(N+1)
eyy2=fltarr(N+1)
eyy3=fltarr(N+1)
xstep=fltarr(N+1)
ystep=fltarr(N+1)
y0=fltarr(Nstep+1,121)
y=fltarr(n1,Nstep+1,121)
y=fltarr(n1,Nstep+1,121)
image1=bytarr(300,300)
image2=bytarr(300,300)
image3=bytarr(200,200)
image4=bytarr(500,300)
animationA=bytarr(300,300,Nstep+1)
animationB=bytarr(200,200,Nstep+1)
t=findgen(N+1)+samplesa
animationAB=bytarr(500,300,Nstep+1)
```

The isopleths are calculated with the help of linear beam theory.

\[ e(x, y) = e_0(x, y) + Kx(x, y)x + Ky(x, y)y \]

In order to determine the values of \( Kx, Ky \), and \( e_0 \), 3 sets of rosette coordinates and their values of longitudinal normal strain \( e_y \) are used.

\( (x1, y1) = \text{medial}, (x2, y2) = \text{anterior}, (x3, y3) = \text{lateral} \) are read from

Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.
; base="_coord.dat". Eyy1, eyy2, eyy3 are respectively the normal strains at
; medial, anterior, and lateral rosette

; eyy1=eyy(0,samplesa:samplesp)
eyy2=eyy(1,samplesa:samplesp)
eyy3=eyy(2,samplesa:samplesp)
loadr,s
status=dc_read_tiff('/homes/stefan/pvwave/mauricio/image'+base+'/plot.tif', image1)

; Calculation of the parameters and the selected isopleth (neutral axis y0,
tension isopleths y, and compression isopleths yn).

\[ Ky = \frac{(eyy1-eyy2)*(x2-x3)-(eyy2-eyy3)*(x1-x2)}{KLFF} \]
\[ Kx = \frac{(eyy1-eyy2)-Ky*(y1-y2)}{(x1-x2)} \]
eO=eyy1-Kx*x 1-Ky*y 1

for j=1,n1 do begin
for i=0,N,step do begin
Istep=i/step
yO-l,Istep,*E j*50-eO(i)-Kx(i)*x )/Ky(i)
yO(Istep,*H O-eO(i)-Kx(i)*x )Ky(i)
yn('j-1,Istep,*)=(-j*50-eO(i)-Kx(i)*x )Ky(i)
endfor
endfor

; The animation consists of 4 parts:
; window,1: animationA of cross-section with isopleths
; window,2: animationB of anterior normal strain in time
; window,3: assembly of prior animations
; window,4: final animationAB

window,1,pixmap,xsize=300,ysize=300
for i=0,Nstep do begin
a=224
b=48
number=samplesa+(i*step)
plot.findgen(2),/nodata,xrange=[-6,6],yrange=[-6,6]
tv,image1
for j=1,n1 do begin
plot,x,y(j-1,1,*),color=a
plot,x,y0(i,*)color=168,pSYM=5
plot,x,yn(j-1,1,*),color=b
xyout,70,90,'sample:',device,color=244
xyout,50,80,number,device,color=244
a=a-16
b=b-10
endfor
image2=wrd(0,0,300,300)
aminationA(*,*,i)=image2
endfor

window,2,pixmap,xsize=200,ysize=200

Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.
plot,t,eyy2
oploto(t(0:1),eyy2(0:1),color=132,psym=6
image3=vrd(0,0,200,200)
animationB(*,*,0)=image3
for i=step,N,step do begin
  Istep=i/step
  oplot(t(i-1:i),eyy2(i-1:i),color=132,psym=6
  image3=vrd(0,0,200,200)
animationB(*,*,Istep)=image3
endfor
wdelete,2
window,3,/pixmap,xsize=500,ysize=300
xyouts,330,210,/device,'strain v.s.time at anterior gage'
for i=0,Nstep do begin
tv,animationA(*,*,i),0,0
tv,animationB(*,*,i),300,0
image4=vrd(0,0,500,300)
animationAB(*,*,i)=reverse(image4,0)
endfor
wdelete,3
label_1:
  window,4,xsize=500,ysize=300,xpos=100,ypos=100,$
title='longitudinal normal strain animation'
  movie,animationAB,order=1

; The animation can be stopped at a certain samplepoint by pressing "q".
; After stopping the animation there are two possibilities:
; 1) Returning to the animation, press an arbitrary key twice.
; 2) Leaving the program, by pressing "s" twice

message=END OF PROGRAMME: PRESS 2*"S", RETURN TO ANIMATION; PRESS 2* ANY KEY!
HAK,mesg=message
if (get_kbdr(1) ne 's') then goto,label_1
wdelete,4
end

---

Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.
pro coord, base

coord="/homes/stefan/pvwave/maurice/data"+base="/gradcrd.dat"
centr=fltarr(2)
new_image=bytarr(500,500)
;
; Program for interactively determining the coordinates of the strain gages. The coordinates are then saved in file "gradcrd.dat"
;
; fastshct, 15
window, 0, xsize=250, ysize=250, xpos=100, ypos=100,$
title="determining centroid.....";
status=dc_read_tiff(/homes/stefan/pvwave/maurice/image"+base"+_mini.tif,$ mini)
mini(where (mini lt 100))=0
mini(where (mini gt 100))=1
fitness, mini
centroid, mini, centr
;print, centr(0), centr(1)
new_image=rebin(mini, 500, 500)
delete, 0
window, 1, xsize=500, ysize=500, xpos=100, ypos=100,$
title="point coordinates .....";
TV, new_image
;
; interactively reading the coordinates of the rosettes by clicking the markings on the bone surface in the order: M, A, L
;
cursor, x1, y1, 3, /device
print,'medial:', x1/2, y1/2
cursor, x2, y2, 3, /device
print,'anterior:', x2/2, y2/2
cursor, x3, y3, 3, /device
print,'lateral:', x3/2, y3/2
delete, 1
openw, 1, coord
printf, 1, x1/2, y1/2, x2/2, y2/2, x3/2, y3/2, centr(0), centr(1)
close, 1
end
Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.
eyy1=eyy(0,samplesa: samplesp)
eyy2=eyy(1,samplesa: samplesp)
eyy3=eyy(2,samplesa: samplesp)

\[ \begin{align*}
K_y &= (eyy_1 - eyy_2) * (x_2 - x_3) + (eyy_2 - eyy_3) * (x_1 - x_2) / y \\
K_x &= (eyy_1 - eyy_2 - K_y * (y_1 - y_2)) / (x_1 - x_2)
\end{align*} \]

traceconta, bone, ex, ey, px, py, n=24, an=0, cx, cy

; calculation of the strain as a 3D function

for i=0,m do begin
e1(i,*)=e0+Kx*px(i)+ky*py(i)
e2(i,*)=e0+Kx*ex(i)+ky*ey(i)
endfor

; convention for calculating strain gradients is clockwise, this is done by counting from position m to 0.

for i=m-1,0,-1 do begin
d1(i)=sqrt((px(i+1)-px(i))^2+(py(i+1)-py(i))^2)
d2(i)=sqrt((ex(i+1)-ex(i))^2+(ey(i+1)-ey(i))^2)
gx1(i)=(px(i+1)+px(i))/2
gy1(i)=(py(i+1)+py(i))/2
gx2(i)=(ex(i+1)+ex(i))/2
gy2(i)=(ey(i+1)+ey(i))/2
for j=0,N,step do begin
    Jstep=j/step
egr1(i,Jstep)=10*(e1(i+1,j)-e1(i,j))/d1(i)
egr2(i,Jstep)=10*(e2(i+1,j)-e2(i,j))/d2(i)
vert_x(0:23,*)=0
vert_z(24:47,Jstep)=egr1(*,Jstep)
hor_x(0:23,Jstep)=egr2(*,Jstep)
endfor
endfor

vert_x(0:23)=gx1
vert_x(24:47)=gx1
vert_y(0:23)=gy1
vert_y(24:47)=gy1
hor_x(0:23)=gx1
hor_x(24:47)=gx2
hor_y(0:23)=gy1
hor_y(24:47)=gy2

window,1/pixmap,xsize=500,ysize=300,xpos=600,ypos=100
t5d,reset,translate=[200,0,0],scale=[0.98,0.42,1],rotate=[0,0,10]
for j=0,Nstep do begin
    number=samplesa+(j*step)
surface,findgen(2,2)/nodata/save/data,zrange=[z_min,z_max],$
xrange=[0,250],yrange=[0,250]

Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.
for i=m-1,0,-1 do begin
if (egr1(i,j) gt 90) then a=255
if ((egr1(i,j) lt 80) and (egr1(i,j) gt 60)) then a=192
if ( (egr1(i,j) lt 60) and (egr1(i,j) gt 40)) then a=174
if ( (egr1(i,j) lt 40) and (egr1(i,j) gt 20)) then a=142
if ( (egr1(i,j) lt 20) and (egr1(i,j) gt 0)) then a=110
if ( (egr1(i,j) lt 0) and (egr1(i,j) gt -20)) then a=96
if ( (egr1(i,j) lt -20) and (egr1(i,j) gt -40)) then a=84
if ( (egr1(i,j) lt -40) and (egr1(i,j) gt -60)) then a=48
if ( (egr1(i,j) lt -60) and (egr1(i,j) gt -80)) then a=64
if (egr1(i,j) lt -80) then a=32
plots, [vert_x(i), vert_x(i+24)], [vert_y(i), vert_y(i+24)],
[vert_z(i), vert_z(i+24)], $t3d/data, psym=1, color=a
plots, [hor_x(i), hor_x(i+24)], [hor_y(i), hor_y(i+24)],
[hor_z(i), hor_z(i+24)], $t3d/data, psym=1, color=130
endfor
plots, gx1, gy1, 0/t3d/data, psym=5, color=142
plots, gx2, gy2, 0/t3d/data, psym=5, color=142
plots, gx1, gy1, egr1(*, j)/t3d/data, psym=5, color=130
plots, gx2, gy2, egr2(*, j)/t3d/data, psym=5, color=130
xyouts, 410, 280, 'Sample', /device
xyouts, 430, 280, number, /device
image = tvrd(0, 0, 500, 300)
animationA(*, *, j) = image
endfor
wdelete, 1
window, 2/GPL pixmap, xsize=200, ysize=200
plot, t, eyy2
oplot, t(0:1), eyy2(0:1), color=110, psym=6
image3 = tvrd(0, 0, 200, 200)
animationB(*, *, 0) = image3
for i=step, N, step do begin
Istep /= step
oplot, t(i-1:i), eyy2(i-1:i), color=132, psym=6
image3 = tvrd(0, 0, 200, 200)
animationB(*, *, istep) = image3
endfor
wdelete, 2
loadct, 5
window, 3/GPL pixmap, xsize=700, ysize=300
xyouts, 20, 210, /device, 'strain v.s.time at anterior gage', color=244
xyouts, 20, 285, /device, ' > 80 uE', color=244
xyouts, 20, 275, /device, ' 80 < uE < 60', color=192
xyouts, 20, 265, /device, ' 60 < uE < 40', color=174
xyouts, 20, 255, /device, ' 40 < uE < 20', color=142
xyouts, 20, 245, /device, ' 20 < uE < 0', color=110
xyouts, 110, 285, /device, ' 0 < uE < -20', color=96
xyouts, 100, 275, /device, '-20 < uE < -40', color=84
xyouts, 100, 265, /device, '-40 < uE < -60', color=48
xyouts, 100, 255, /device, '-60 < uE < -80', color=64
xyouts, 100, 245, /device, '< -80 uE', color=32
for i=0, N, step do begin
tvanimationA(*, *, i), 200, 0
tvanimationB(*, *, i), 0, 0
endfor

Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.
image4=rvrd(0,0,700,300)
animationAB(*,*;)=image4
endfor
wdelete,3
label_1:
  window,4,xsize=700,ysize=300,xpos=360,ypos=100,$
  title='strain gradients of longitudinal normal strain'
  movie,animationAB,order=0
;
; The animation can be stopped at a certain samplepoint by pressing "q".
; After stopping the animation there are two possibilities:
; 1) Returning to the animation, press an arbitrary key twice.
; 2) Leaving the program, by pressing "s" twice.
message=' END OF PROGRAMME: PRESS 2* "S", RETURN TO ANIMATION: PRESS 2* ANY KEY!'
HAK,mesg=message
if (get_kbrd(1) ne 's') then goto, label_1
wdelete,4
end

Analysis of in vivo Recorded Bone Strains using the Command Language PV-WAVE.