Introduction to Diffusion Tensor Imaging Mathematics: Part II. Anisotropy, Diffusion-Weighting Factors, and Gradient Encoding Schemes

PETER B. KINGSLEY

Department of Radiology, North Shore University Hospital, 300 Community Drive, Manhasset, New York 11030, and Department of Radiology, New York University School of Medicine, New York, New York

ABSTRACT: The mathematical aspects of diffusion tensor magnetic resonance imaging (DTMRI, or DTI), the measurement of the diffusion tensor by magnetic resonance imaging (MRI), are discussed in this three-part series. In part II, different formulas for measuring diffusion anisotropy are compared in two and three dimensions. The use of magnetic field gradients to measure diffusion is explained, including calculation of the diffusion-weighting *b* factor. Selection of gradient directions is discussed, with the focus on directions defined by the regular polyhedra (Platonic solids). © 2006 Wiley Periodicals, Inc. Concepts Magn Reson Part A 28A: 123–154, 2006

KEY WORDS: anisotropy; *b* factor; diffusion; diffusion gradient scheme; diffusion tensor; DTI; icosahedron; tetrahedron

INTRODUCTION

This three-part series discusses the mathematical aspects of diffusion tensor magnetic resonance imaging (DTMRI, or DTI). Part I presented formulas for rotation of vectors and tensors and explained how to calculate the tensor's eigenvectors, which correspond to the directions of the diffusion ellipsoid's axes, and eigenvalues, which correspond to the squares of the lengths of the diffusion ellipsoid's hemiaxes.

This second part discusses an important piece of information in the tensor and explains more about the acquisition of DTI data. Part II begins with a discussion of different ways to evaluate the degree of anisotropy of a tensor, which is commonly expressed as a diffusion anisotropy index (DAI). The degree of anisotropy, as reflected in DAIs, is important because it has been found to be decreased in one or more brain regions in a number of diseases. Calculation of the diffusion-weighting *b* factor is explained next. Meth-

Received 12 September 2005; revised 12 December 2005; accepted 12 December 2005

Correspondence to: Peter Kingsley: E-mail: pkingsle@nshs.edu

Concepts in Magnetic Resonance Part A, Vol. 28A(2) 123–154 (2006)

Published online in Wiley InterScience (www.interscience.wiley.com). DOI 10.1002/cmr.a.20049

ods for eliminating the effects of nondiffusion gradients in diffusion-weighted imaging (DWI) and DTI, both static background gradients and imaging gradients, are presented. Although these methods are not applied universally, they will become more important at higher field strengths (3 Tesla and beyond), especially for tractography. The last section explores different gradient encoding schemes for DWI and DTI. Knowledge about gradient encoding schemes allows the selection of an optimal scheme for a DTI study and the evaluation of schemes used in other studies.

Part III explains how the tensor is calculated from DWI data in six or more directions, discusses computer simulations and the effects of noise on DTI measurements, and explores the optimization of DTI data acquisition and processing.

DIFFUSION ANISOTROPY INDICES

The purposes of this section are (*1*) to show how information about the degree of anisotropy in the diffusion ellipsoid, as represented by the tensor, can be quantified in a single number called a diffusion anisotropy index (DAI), (*2*) to show how to calculate several proposed DAIs, (*3*) to compare the properties of these DAIs, and (*4*) to show that the commonly used DAIs have similar abilities to tell whether two tensors have the same or different degrees of anisotropy.

Background

The full diffusion tensor **D** contains much information. Many applications require a simple idea of the amount of anisotropy. Some two-dimensional plots have been suggested for this purpose (*1*, *2*). However, for quantitative measurements in brain white matter, a single number is more useful for comparing different groups or for monitoring changes over time. Such a single number is called a "diffusion anisotropy index" or DAI, and several have been suggested. The most widely used DAIs range from 0 (isotropic) to 1 (anisotropic) for non-negative eigenvalues, and they can be calculated from the tensor elements without diagonalization (calculation of eigenvalues).

Trying to understand the most commonly used DAIs by reading published articles can be confusing for several reasons:

1. Formulas may be expressed in terms of diffusion tensor elements, eigenvalues, or rotational invariants, and the conversions between these forms are not always obvious. The equivalent formulas are all shown in Tables 1–4.
2. Some formulas may be expressed in different ways by the use of algebraic identities (*3*) such as

$$3[(D_{xx} - D_{av})^2 + (D_{yy} - D_{av})^2 + (D_{zz} - D_{av})^2] = (D_{xx} - D_{yy})^2 + (D_{yy} - D_{zz})^2 + (D_{zz} - D_{xx})^2 \quad [1]$$

$$3[(\lambda_1 - D_{av})^2 + (\lambda_2 - D_{av})^2 + (\lambda_3 - D_{av})^2] = (\lambda_1 - \lambda_2)^2 + (\lambda_2 - \lambda_3)^2 + (\lambda_3 - \lambda_1)^2 \quad [2]$$

and the corresponding 2D identities

$$2[(D_{xx} - D_{av})^2 + (D_{yy} - D_{av})^2] = (D_{xx} - D_{yy})^2 \quad [3]$$

$$2[(\lambda_1 - D_{av})^2 + (\lambda_2 - D_{av})^2] = (\lambda_1 - \lambda_2)^2 \quad [4]$$

3. Some DAIs have been redefined to range from 0 for isotropic diffusion to 1 for completely anisotropic diffusion. Originally fractional anisotropy (*FA*) ranged from 0 to 1, relative anisotropy (*RA*) from 0 to $2^{1/2}$, and volume ratio (*VR*) from 1 to 0. Dividing *RA* by $2^{1/2}$ so that it ranges from 0 to 1 may produce inconsistent formulas if it is not renamed (*2, 4–7*). Renaming the rescaled *RA* as A_σ (*8*) or A_{sd} (*9*) may appear to create an entirely new DAI. Similarly, using $1 - VR$ so that it ranges from 0 to 1 produces inconsistent formulas if it is not renamed (*5, 10*), and a new apparent DAI when it is renamed volume fraction, *VF* (*2*). This article uses the notation *sRA* to indicate *RA* scaled to be between 0 and 1.
4. Mistakes have occurred in published formulas of *FA* (*9, 11, 12*), A_{sd} (*9, 12–14*), *VR* (*15*), and *VF* (*2*).

Intravoxel Diffusion Anisotropy Indices in Three Dimensions

The DAI formulas for standard 3D diffusion are expressed in terms of eigenvalues in Table 1, and in

Table 1 Summary of 3D Intravoxel DAI Formulas That Range from Zero (Isotropic) to One (Anisotropic), Expressed in Terms of Eigenvalues

Index	Definition	Eigenvalue Formula
Fractional Anisotropy (*FA*)	$(3\mathbf{D}_{an}:\mathbf{D}_{an}/2\mathbf{D}:\mathbf{D})^{1/2} = (1 - I_2/I_4)^{1/2}$	$\sqrt{\frac{3[(\lambda_1 - D_{av})^2 + (\lambda_2 - D_{av})^2 + (\lambda_3 - D_{av})^2]}{2(\lambda_1^2 + \lambda_2^2 + \lambda_3^2)}}$
Scaled relative anisotropy (*sRA*)	$(\mathbf{D}_{an}:\mathbf{D}_{an}/6)^{1/2}/D_{av} = (1 - D_{surf}^2)^{1/2}/D_{av} = (1 - 3I_2/I_1^2)^{1/2}$	$\frac{\sqrt{(\lambda_1 - D_{av})^2 + (\lambda_2 - D_{av})^2 + (\lambda_3 - D_{av})^2}}{\sqrt{6}\,D_{av}}$
Volume fraction (*VF*), volume ratio (*VR*)	$1 - I_3/D_{av}^3 = 1 - (D_{vol}/D_{av})^3 = 1 - 27I_3/(I_1)^3$	$1 - \lambda_1\lambda_2\lambda_3/D_{av}^3$
UA_{surf}	$1 - D_{surf}/D_{av}$	$1 - [(\lambda_1\lambda_2 + \lambda_2\lambda_3 + \lambda_3\lambda_1)/3]^{1/2}/D_{av}$
UA_{vol}	$1 - D_{vol}/D_{av}$	$1 - (\lambda_1\lambda_2\lambda_3)^{1/3}/D_{av}$
$UA_{vol,surf}$	$1 - D_{vol}/D_{surf} = (UA_{vol} - UA_{surf})/(UA_{surf} - 1)$	$1 - (\lambda_1\lambda_2\lambda_3)^{1/3}/[(\lambda_1\lambda_2 + \lambda_2\lambda_3 + \lambda_3\lambda_1)/3]^{1/2}$
Lattice index, LI_N	$(FA + FA^2)/2$	

Note: $\mathbf{D}_{an}$ is the anisotropic part of $\mathbf{D}$ (Eq. [I-67]). The rotational invariants I_1, I_2, I_3, D_{av}, D_{surf}, and D_{vol} were defined in Part I.

terms of tensor elements in Table 2. The most widely used DAI is probably *FA* (*10*, *16–20*). Although *RA* and *VR* were introduced at the same time as *FA*, the popularity of *FA* may stem from the fact that it ranges from 0 for isotropic diffusion to 1 for completely anisotropic diffusion, whereas *RA* ranges from 0 to $2^{1/2}$, and *VR* ranges from 1 to 0. However, *sRA* appears to have some advantage in that for cylindrical symmetry, its value is a linear function of the largest eigenvalue (*5*), whereas *FA* is concave down and *VF* has a sigmoid shape. Nevertheless, the initial preference for *FA* appears to persist, perhaps because some people have a sense of the expected *FA* values in different brain regions, or they wish to compare new studies with older published data.

In addition to these three common DAIs, other indices have been proposed. Armitage and Bastin derived the gamma-variate anisotropy index (*GV*) from *sRA* in an attempt to improve the "sensitivity" over the *RA* range found in the brain (*9*). Bahn pro-

Table 2 Summary of 3D Intravoxel DAI Formulas That Range from 0 (Isotropic) to 1 (Anisotropic), Expressed in Terms of Tensor Elements

Index	Tensor Element Formula
Fractional anisotropy (*FA*)	$\sqrt{\frac{3[(D_{xx} - D_{av})^2 + (D_{yy} - D_{av})^2 + (D_{zz} - D_{av})^2 + 2(D_{xy}^2 + D_{xz}^2 + D_{yz}^2)]}{2[D_{xx}^2 + D_{yy}^2 + D_{zz}^2 + D_{zz}^2 + 2(D_{xy}^2 + D_{xz}^2 + D_{yz}^2)]}}$
Scaled relative anisotropy (*sRA*)	$\frac{\sqrt{(D_{xx} - D_{av})^2 + (D_{yy} - D_{av})^2 + (D_{zz} - D_{av})^2 + 2(D_{xy}^2 + D_{xz}^2 + D_{yz}^2)}}{\sqrt{6}\,D_{av}}$
Volume fraction (*VF*), Volume ratio (*VR*)	$1 - [D_{xx}D_{yy}D_{zz} + 2D_{xy}D_{xz}D_{yz} - (D_{zz}D_{xy}^2 + D_{yy}D_{xz}^2 + D_{yy}D_{xz}^2 + D_{xx}D_{yz}^2)]/D_{av}^3$
UA_{surf}	$1 - \frac{\sqrt{D_{xx}D_{yy} + D_{yy}D_{zz} + D_{zz}D_{xx} - (D_{xy}^2 + D_{xz}^2 + D_{yz}^2)}}{\sqrt{3}\,D_{av}}$
UA_{vol}	$1 - \sqrt[3]{D_{xx}D_{yy}D_{zz} + 2D_{xy}D_{xz}D_{yz} - (D_{zz}D_{xy}^2 + D_{yy}D_{xz}^2 + D_{xx}D_{yz}^2)}D_{av}$
$UA_{vol,surf}$	$1 - \frac{\sqrt[3]{D_{xx}D_{yy}D_{zz} + 2D_{xy}D_{xz}D_{yz} - (D_{zz}D_{xy}^2 + D_{yy}D_{xz}^2 + D_{xx}D_{yz}^2)}}{\sqrt{\frac{D_{xx}D_{yy} + D_{yy}D_{zz} + D_{zz}D_{xx} - (D_{xy}^2 + D_{xz}^2 + D_{yz}^2)}{3}}}$
Gamma variate	$[ae^{-bsRA}(b^2sRA^2 + 2bsRA + 2) - 2a]/b^3$, $a = -259.57$, $b = 8$
Lattice index, LI_N	$(FA + FA^2)/2$

posed some new rotational invariants and new DAIs derived from those invariants. Some of these proposed DAIs were identical to previously defined DAIs (*1*). Ulug et al. introduced some new rotational invariants (D_{av}, D_{surf}, D_{vol}, and D_{mag}) derived from the three fundamental invariants I_1, I_2, and I_3 (*21*). They defined three new "ultimate anisotropy" indices—UA_{surf}, UA_{vol}, and $UA_{vol,surf}$—in terms of their newly defined rotational invariants, and showed how to calculate *FA*, *RA*, and *VR* from these same invariants. These new *UA* indices were defined confusingly in the form of $[(-X)^2]^{1/2}$, which equals *X*. The definitions in Tables 1 and 2 avoid this confusion by avoiding the squares and square roots. The *UA* indices all range from 0 for isotropic diffusion to 1 for completely anisotropic diffusion. Ulug et al. (*21*) pointed out that DAIs like theirs and *GV* do not improve sensitivity over *sRA* or *FA* because their algebraic formulas affect signal and noise in the same way. All these DAIs are included in Tables 1 and 2.

The different DAIs do not represent independent descriptions of anisotropy (*22*). Fractional anisotropy (*6*), UA_{surf}, and *GV* can be expressed in terms of *sRA*, UA_{vol} in terms of *VF*, and $UA_{vol,surf}$ in terms of *sRA* and *VF*.

$$FA = sRA[3/(2sRA^2 + 1)]^{1/2} \quad [5]$$

$$UA_{surf} = 1 - (1 - sRA^2)^{1/2} \quad [6]$$

$$GV = 259.57[1 - e^{-8sRA}(32sRA^2 + 8sRA + 1)]/256 \quad [7]$$

$$sRA = FA/(3 - 2FA^2)^{1/2} = [1 - (1 - UA_{surf})^2]^{1/2} \quad [8]$$

$$VF = 1 - (1 - UA_{vol})^3 \quad [9]$$

$$UA_{vol} = 1 - (1 - VF)^{1/3} \quad [10]$$

$$UA_{vol,surf} = 1 - (1 - VF)^{1/3}/(1 - sRA^2)^{1/2} \quad [11]$$

With cylindrically symmetric diffusion there are two identical eigenvalues, which we call *D*2, and a third eigenvalue, which we call *D*1. In this case the anisotropy can be represented by the parameter *A*, which was defined in (*8*) and is identical to $A_{fiber} = -A_{disk}$ of (*21*):

$$A = (D1/D_{av} - 1)/2 = (D1 - D2)/3D_{av} \quad [12]$$

$$D1 = D_{av}(1 + 2A) \quad [13]$$

$$D2 = D_{av}(1 - A) \quad [14]$$

$$D_{av} = (D1 + 2D2)/3 \quad [15]$$

A ranges from −0.5 for completely anisotropic oblate diffusion ($D1 = 0$) through 0 for isotropic diffusion ($D1 = D2$) to 1 for completely anisotropic prolate diffusion ($D2 = 0$). For cylindrical symmetry, each DAI can be expressed in terms of *A*. For $A > 0$:

$$sRA = A \quad [16]$$

$$FA = A[3/(2A^2 + 1)]^{1/2} \quad [17]$$

$$GV = 259.57[1 - e^{-8A}(32A^2 + 8A + 1)]/256 \quad [18]$$

$$UA_{surf} = 1 - (1 - A^2)^{1/2} \quad [19]$$

$$VF = 3A^2 - 2A^3 \quad [20]$$

$$UA_{vol} = 1 - [1 - (3A^2 - 2A^3)]^{1/3} \quad [21]$$

$$UA_{vol,surf} = 1 - [1 - (3A^2 - 2A^3)]^{1/3}/(1 - A^2)^{1/2} \quad [22]$$

For $A < 0$, it is necessary to replace *A* with $-A$ in the formulas for *sRA*, *FA*, and *GV*:

$$sRA = -A \quad (A < 0) \quad [23]$$

$$FA = -A[3/(2A^2 + 1)]^{1/2} \quad (A < 0) \quad [24]$$

$$GV = 259.57[1 - e^{8A}(32A^2 - 8A + 1)]/256 \quad (A < 0) \quad [25]$$

The seven DAIs in Tables 1 and 2, Eqs. [5–11], and Eqs. [16–25] are plotted as a function of *A* in Fig. 1. The derivative of each DAI with respect to *A* can be calculated as follows for $A > 0$:

$$dsRA/dA = 1 \quad [26]$$

$$dFA/dA = [3/(2A^2 + 1)^3]^{1/2} \quad [27]$$

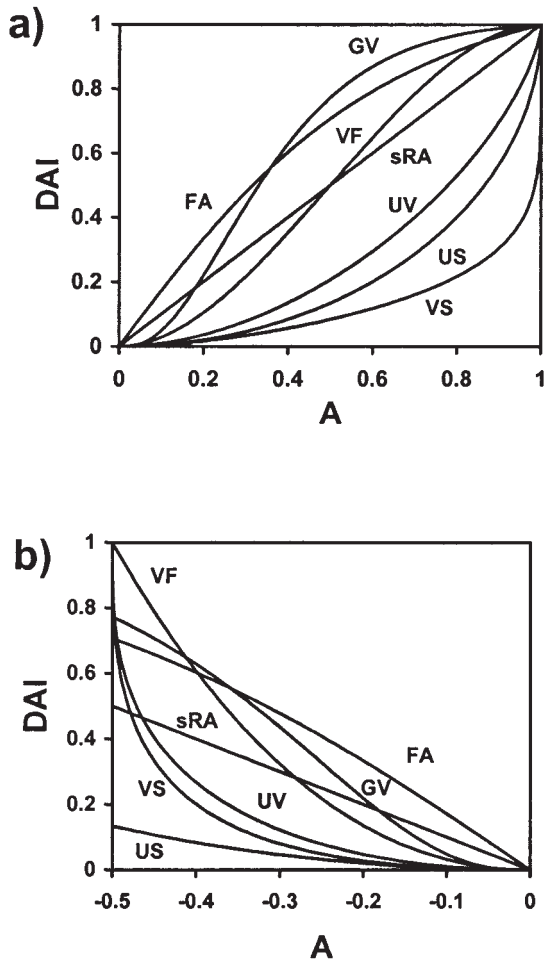

Figure 1 Values of 3D diffusion anisotropy indices (DAIs) as a function of the cylindrical symmetry parameter A (Eq. [12]), for (a) prolate ellipsoids ($A > 0$), and (b) oblate ellipsoids ($A < 0$).

$$dGV/dA = 259.57A^2e^{-8A} \quad [28]$$

$$dUA_{surf}/dA = A/(1 - A^2)^{1/2} \quad [29]$$

$$dVF/dA = 6A(1 - A) \quad [30]$$

$$dUA_{vol}/dA = 2A(1 - A)/[1 - (3A^2 - 2A^3)]^{2/3} \quad [31]$$

$$\frac{dUA_{vol,surf}}{dA} = \frac{A(1 - A)^{1/2}}{(1 + A)^{3/2}[1 - (3A^2 - 2A^3)]^{2/3}} \quad [32]$$

For $A < 0$, the formulas for the derivatives of *sRA*, *FA*, and *GV* must be modified as follows:

$$dsRA/dA = -1 \quad (A < 0) \quad [33]$$

$$dFA/dA = -[3/(2A^2 + 1)^3]^{1/2} \quad (A < 0) \quad [34]$$

$$dGV/dA = -259.57A^2e^{8A} \quad (A < 0) \quad [35]$$

These derivatives are shown in Fig. 2. Equations [5–11] also lead to the following relationships for the derivatives with respect to *sRA* or UA_{vol}: *dFA/dsRA* (*7*), *dGV/dsRA* (*9*), $dUA_{surf}/dsRA$ (*22*), and *dVF/*dUA_{vol} (*22*).

$$dFA/dsRA = (FA/sRA)^3/3 = [3/(2sRA^2 + 1)^3]^{1/2} \quad [36]$$

$$dGV/dsRA = 259.57sRA^2e^{-8sRA} \quad [37]$$

$$dUA_{surf}/dsRA = sRA/(1 - sRA^2)^{1/2} \quad [38]$$

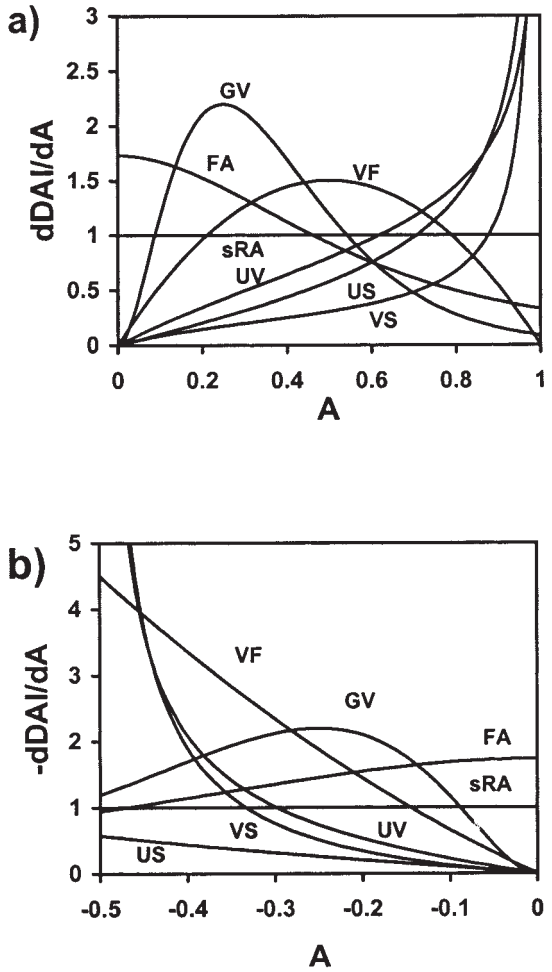

Figure 2 Derivatives of 3D diffusion anisotropy indices (DAIs) with respect to the cylindrical symmetry parameter A (Eq. [12]), for (a) prolate ellipsoids ($A > 0$), and (b) oblate ellipsoids ($A < 0$).

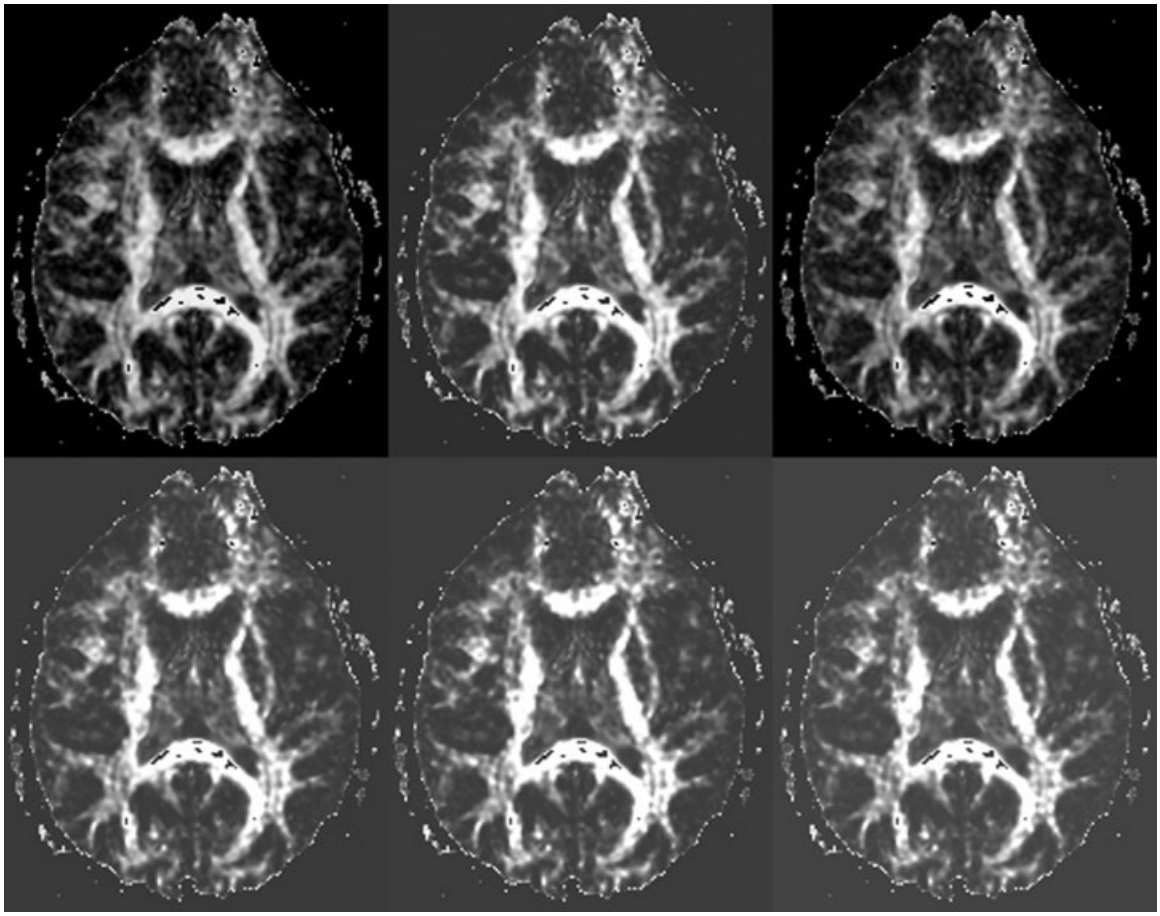

Figure 3 Anisotropy maps for six of the 3D DAIs, excluding UA_{surf}. Top row: *FA*, *GV*, and *sRA*. Bottom row: *VR*, UA_{vol}, and $UA_{vol,surf}$. The linear intensity scale is different for each image. The DTI data were acquired with six different gradient directions corresponding to the edges of a cube (six gradient pairs). Each direction was repeated twice with positive gradients and twice with negative gradients to eliminate cross-terms between the diffusion gradients and imaging gradients (*44*). The black spots in the splenium of the corpus callosum are areas where the eigenvalue calculation failed, presumably due to the presence of negative eigenvalues.

$$dVF/dUA_{vol} = 3(1 - UA_{vol})^2 \qquad [39]$$

An image whose pixel intensity is proportional to the value of a DAI is called a DAI map. DAI maps typically are scaled by a factor of 1,000 so that they have pixel intensities comparable to those of the raw DW images. Examples of DAI maps for six of the DAIs discussed here, excluding UA_{surf}, are shown in Fig. 3. These images were acquired in a volunteer who gave informed consent under a protocol approved by the Institutional Review Board of the North Shore Long Island Jewish Health System. The intensity scale is linear in each case, but the scale is different in each DAI map. Notice that all six DAI maps can have the display range adjusted so that the same key features are visible in each one. However, there are also noticeable differences with the display scales used here. The maps could be made to look even more similar with nonlinear display scales. These DAI maps show that the same basic information is available with all the DAIs, though certain features may be seen more easily with one DAI than with another.

Unnormalized DAIs

All the DAIs discussed so far have been dimensionless. Each DAI has a numerator, which contains information about anisotropy, a denominator, which has the same units as the numerator and provides a normalization factor like D_{av}, and a scale factor to ensure that the DAI has a value between 0 and 1 for positive eigenvalues. It has been suggested that sometimes it may be useful to consider the numerator and denom-

Table 3 Summary of 2D Intravoxel DAI Formulas That Range from 0 (Isotropic) to 1 (Anisotropic), Expressed in Terms of Eigenvalues

Index	Definition	Eigenvalue Formula
FA	$(2\mathbf{D}_{an}:\mathbf{D}_{an}/\mathbf{D}:\mathbf{D})^{1/2}$	$\sqrt{\frac{2[(\lambda_1 - D_{av})^2 + (\lambda_2 - D_{av})^2]}{(\lambda_1^2 + \lambda_2^2)}} = (\lambda_1 - \lambda_2)/(\lambda_1^2 + \lambda_2^2)^{1/2}$
sRA	$(\mathbf{D}_{an}:\mathbf{D}_{an}/2)^{1/2}/D_{av}$	$\frac{\sqrt{(\lambda_1 - D_{av})^2 + (\lambda_2 - D_{av})^2}}{\sqrt{2}\,D_{av}} = (\lambda_1 - \lambda_2)/(\lambda_1 + \lambda_2)$
VF	$1 - I_2/D_{av}^2$	$1 - \lambda_1\lambda_2/D_{av}^2$
UA_{area}	$1 - D_{area}/D_{av}$	$1 - 2(\lambda_1\lambda_2)^{1/2}/(\lambda_1 + \lambda_2)$
LI_N	$(FA + FA^2)/2$	

Note: $\mathbf{D}_{an}$ is the anisotropic part of $\mathbf{D}$ (Eq. [I-67]). The rotational invariants I_1, I_2, I_3, D_{av}, and D_{area} were defined in Part I.

inator separately to determine whether a change in a DAI is caused primarily by a change in the anisotropic numerator or the isotropic denominator (*23*). The numerator of *sRA*, scaled to be no greater than D_{av} for positive eigenvalues, could be called the absolute anisotropy, *AA*:

$$AA = sRA*D_{av} = \{[(\lambda_1 - D_{av})^2 + (\lambda_2 - D_{av})^2 + (\lambda_3 - D_{av})^2]/6\}^{1/2} \quad [40]$$

$$AA = \{[(D_{xx} - D_{av})^2 + (D_{yy} - D_{av})^2 + (D_{zz} - D_{av})^2 + 2D_{xy}^2 + 2D_{xz}^2 + 2D_{yz}^2]/6\}^{1/2} \quad [41]$$

One advantage of *AA* over *sRA* or *FA* is that it can be calculated even if the $b = 0$ signal intensity is not known, or is inaccurate or imprecise. For example, if DTI data were collected for calculation of eigenvectors, where no $b = 0$ knowledge is required, and later some anisotropy information were desired, *AA* could provide some anisotropy information.

Intravoxel Diffusion Anisotropy Indices in Two Dimensions

The DAI formulas for 2D diffusion are expressed in terms of eigenvalues in Table 3 and tensor elements in Table 4. As with the 3D DAIs, some 2D DAIs can be expressed in terms of other 2D DAIs.

$$FA = sRA[2/(sRA^2 + 1)]^{1/2} \quad (2D) \quad [42]$$

$$sRA = FA/(2 - FA^2)^{1/2} = [1 - (1 - UA_{area})^2]^{1/2} \quad (2D) \quad [43]$$

$$UA_{area} = 1 - (1 - sRA^2)^{1/2} = 1 - (1 - VF)^{1/2} \quad (2D) \quad [44]$$

$$VF = sRA^2 \quad (2D) \quad [45]$$

Notice that every 2D DAI can be expressed as a function of *sRA*, and therefore as a function of every other 2D DAI. Furthermore, unlike ellipsoids, ellipses do not have prolate and oblate categories. As a result, the 2D analog of *A* is never negative, and is given by

$$A = (D1/D_{av} - 1) = (D1 - D2)/2D_{av} \quad (2D) \quad [46]$$

$$D1 = D_{av}(1 + A) \quad (2D) \quad [47]$$

$$D2 = D_{av}(1 - A) \quad (2D) \quad [48]$$

Table 4 Summary of 2D Intravoxel DAI Formulas That Range from 0 (Isotropic) to 1 (Anisotropic), Expressed in Terms of Tensor Elements

Index	Tensor Element Formula
FA	$\{[(D_{xx} - D_{av})^2 + (D_{yy} - D_{av})^2 + 2D_{xy}^2]/[D_{xx}^2 + D_{yy}^2 + 2D_{xy}^2]\}^{1/2}$
sRA	$\{[(D_{xx} - D_{av})^2 + (D_{yy} - D_{av})^2 + 2D_{xy}^2]/2\}^{1/2}/D_{av}$
VF	$1 - (D_{xx}D_{yy} - D_{xy}^2)/D_{av}^2$
UA_{area}	$1 - 2(D_{xx}D_{yy} - D_{xy}^2)^{1/2}/(D_{xx} + D_{yy})$
LI_N	$(FA + FA^2)/2$

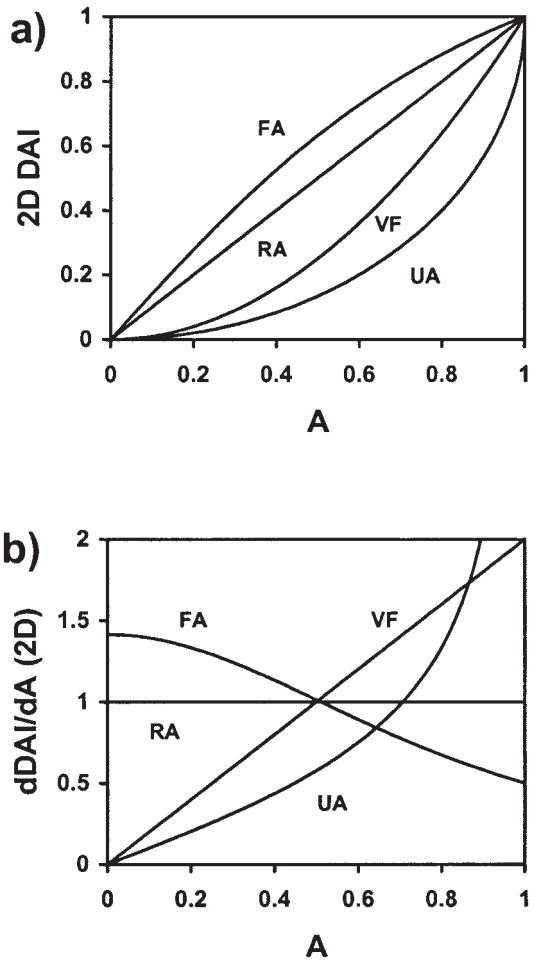

Figure 4 (a) Values of 2D diffusion anisotropy indices (DAIs) as a function of the anisotropy parameter A (Eq. [46]). (b) Derivatives of 2D diffusion anisotropy indices (DAIs) with respect to the anisotropy parameter A.

where $D1 > D2$. The 2D DAIs can also be expressed as a function of A.

$$sRA = A \quad (2D) \qquad [49]$$

$$FA = A[2/(A^2 + 1)]^{1/2} \quad (2D) \qquad [50]$$

$$UA_{area} = 1 - (1 - A^2)^{1/2} \quad (2D) \qquad [51]$$

$$VF = A^2 \quad (2D) \qquad [52]$$

The four DAIs in Tables 3 and 4, Eqs. [42–45], and Eqs. [49–52] are plotted as a function of A in Fig. 4(a). The derivative of each 2D DAI with respect to A can also be calculated.

$$dsRA/dA = 1 \quad (2D) \qquad [53]$$

$$dFA/dA = [2/(A^2 + 1)^3]^{1/2} \quad (2D) \qquad [54]$$

$$dUA_{area}/dA = A/(1 - A^2)^{1/2} \quad (2D) \qquad [55]$$

$$dVF/dA = 2A \quad (2D) \qquad [56]$$

These derivatives are shown in Fig. 4(b). Because $sRA = A$, replacing A with sRA in Eqs. [54–56] yields the derivative with respect to sRA.

$$dFA/dsRA = [2/(sRA^2 + 1)^3]^{1/2} \quad (2D) \qquad [57]$$

$$dUA_{area}/dsRA = sRA/(1 - sRA^2)^{1/2} \quad (2D) \qquad [58]$$

$$dVF/dsRA = 2sRA \quad (2D) \qquad [59]$$

Intervoxel Diffusion Anisotropy Indices

The DAIs discussed above use a single set of DTI data. It is also possible to define DAIs that use more than one set of data. These different data sets may be repetitions of data in an individual voxel or data from surrounding voxels. These indices will be called "intervoxel indices," even when they are applied to repeated data from a single voxel, because they allow the use of data from neighboring voxels when calculating the DAI in one voxel. One option with multiple data sets is simply to average the raw data before calculating the tensor. In general, when multiple sets of data are going to be processed the same way, it is better to average the data as soon as possible in the chain of calculations. For example, it is better to average each quadrature channel individually before calculating the magnitude. If this is not possible, it is better to average the magnitudes before calculating the tensor. If that is not possible, it is better to average the tensors before calculating eigenvalues and eigenvectors. If the presence of multiple data sets allows new processing steps that are not available to individual data sets, then there may be an advantage of the new process over the original processing of averaged data. The rest of this section will assume that data averaging is not performed, and that separate data sets are to be processed in ways that are not available to single data sets.

Sun et al. introduced the directional-correlation weighted *RA*, *DRA* (*24*). The corresponding *DsRA* will be discussed here. The *DsRA* is calculated like *sRA*, but with two tensors, $\mathbf{D}$ and $\mathbf{D}'$. For a 3D tensor,

$$DsRA = (\mathbf{D}_{an}:\mathbf{D}'_{an}/6D_{av}D'_{av})^{1/2} \qquad [60]$$

$$\mathbf{D}_{an}:\mathbf{D}'_{an} = (D_{xx} - D_{av})(D'_{xx} - D'_{av})$$

$$+ (D_{yy} - D_{av})(D'_{yy} - D'_{av}) + (D_{zz} - D_{av})(D'_{zz} - D'_{av}) + 2(D_{xy}D'_{xy} + D_{xz}D'_{xz} + D_{yz}D'_{yz}) \quad [61]$$

$$DsRA = \sqrt{\frac{\sum_{i=1}^{3} (\lambda_i - D_{av})(\lambda'_i - D'_{av})}{6D_{av}D'_{av}}} \quad [62]$$

Sun et al. compared application of *DRA* to multiple data sets in each voxel, as well as using data sets from neighboring voxels (*24*). A directional-correlation weighted *FA*, *DFA* can be defined similarly.

$$DFA = [(3\mathbf{D}_{an}:\mathbf{D}'_{an})/(2\mathbf{D}:\mathbf{D}')]^{1/2} \quad [63]$$

$$\mathbf{D}:\mathbf{D}' = D_{xx}D'_{xx} + D_{yy}D'_{yy} + D_{zz}D'_{zz} + 2(D_{xy}D'_{xy} + D_{xz}D'_{xz} + D_{yz}D'_{yz}) \quad [64]$$

$$DFA = \sqrt{\frac{3\sum_{i=1}^{3} (\lambda_i - D_{av})(\lambda'_i - D'_{av})}{2(\lambda_1\lambda'_1 + \lambda_2\lambda'_2 + \lambda_3\lambda'_3)}} \quad [65]$$

An unscaled version of *DFA*, without the $(3/2)^{1/2}$ factor, has been called A_{dd} (*12*). The same relationships that apply to *sRA* and *FA* (Eqs. [5], [8], and [36]) apply to *DsRA* and *DFA*.

A more complex DAI that uses diffusion tensor information from neighboring voxels along with the information in a given voxel is the lattice index (*LI*), which was introduced in 1996 (*25*). The original formula was incorrect, and the published correction did not include italics, so two groups that have tried to use the *LI* have published different formulas (*4*, *12*). The correct formula for the basic element of the *LI* seems to be

$$LI_N = \sqrt{\frac{3}{8}} \sqrt{\frac{\mathbf{D}_{an} : \mathbf{D}'_{an}}{\mathbf{D} : \mathbf{D}'}} + \frac{3}{4} \frac{\mathbf{D}_{an} : \mathbf{D}'_{an}}{\sqrt{\mathbf{D} : \mathbf{D}} \sqrt{\mathbf{D}' : \mathbf{D}'}} \quad [66]$$

where $\mathbf{D}$ and $\mathbf{D}_{an}$ refer to one acquisition in one voxel, and $\mathbf{D}'$ and $\mathbf{D}'_{an}$ refer to another acquisition or another voxel. The first term in Eq. [66] is *DFA*/2, and the second term is similar to $DFA^2/2$, but is not the same. The *LI* for a single voxel was calculated as a weighted average of LI_N in the eight surrounding voxels in a slice (*25*). If LI_N were calculated for a single data set in a single voxel, so that $\mathbf{D} = \mathbf{D}'$ and $\mathbf{D}_{an} = \mathbf{D}'_{an}$,

$$LI_N = (FA + FA^2)/2 \quad \text{(single voxel)} \quad [67]$$

$$\frac{dLI_N}{dA} = \frac{\sqrt{3}\,(2A + \sqrt{2A^2 + 1})}{2(2A^2 + 1)^2} \quad \text{(single voxel)} \quad [68]$$

The corresponding intervoxel DAIs for 2D calculations are

$$DsRA = (\mathbf{D}_{an}:\mathbf{D}'_{an}/2D_{av}D'_{av})^{1/2} \quad (2D) \quad [69]$$

$$DFA = (2\mathbf{D}_{an}:\mathbf{D}'_{an}/\mathbf{D}:\mathbf{D}')^{1/2} \quad (2D) \quad [70]$$

$$LI_N = \sqrt{\frac{\mathbf{D}_{an} : \mathbf{D}'_{an}}{2\mathbf{D} : \mathbf{D}'}} + \frac{\mathbf{D}_{an} : \mathbf{D}'_{an}}{\sqrt{\mathbf{D} : \mathbf{D}} \sqrt{\mathbf{D}' : \mathbf{D}'}} \quad (2D) \quad [71]$$

The same 2D relationships that apply to *sRA* and *FA* (Eqs. [42], [43], and [57]) apply to *DsRA* and *DFA*. If LI_N were calculated for a single data set in a single voxel, so that $\mathbf{D} = \mathbf{D}'$ and $\mathbf{D}_{an} = \mathbf{D}'_{an}$,

$$\frac{dLI_N}{dA} = \frac{4A + \sqrt{2(A^2 + 1)}}{2(A^2 + 1)^2} \quad (2D, \text{single voxel}) \quad [72]$$

Sometimes it is convenient to have a single formula for both 2D and 3D, expressed in terms of the number of dimensions, N_D. Several formulas are identical for 2D and 3D, including *D*2 (Eqs. [14] and [48]), *sRA* = *A* (Eqs. [16] and [49]), *dsRA*/*dA* (Eqs. [26] and [53]), UA_{surf} and UA_{area} as functions of *A* (Eqs. [19] and [51]), and dUA_{surf}/dA (3D) = dUA_{area}/dA (2D) (Eqs. [29] and [55]). In addition, the following formulas may be useful:

$$A = (D1/D_{av} - 1)/(N_D - 1) = (D1 - D2)/N_D D_{av} \quad [73]$$

$$D1 = D_{av}[1 + (N_D - 1)A] \quad [74]$$

$$FA = sRA\{N_D/[(N_D - 1)sRA^2 + 1]\}^{1/2} \quad [75]$$

$$sRA = FA/[N_D - (N_D - 1)FA^2]^{1/2} \quad [76]$$

$$dFA/dA = \{N_D/[(N_D - 1)A^2 + 1]^3\}^{1/2} \quad [77]$$

$$FA = [N_D\mathbf{D}_{an}:\mathbf{D}_{an}/(N_D - 1)\mathbf{D}:\mathbf{D}]^{1/2} \quad [78]$$

$$sRA = [\mathbf{D}_{an}:\mathbf{D}_{an}/N_D(N_D - 1)\mathbf{D}:\mathbf{D}]^{1/2}/D_{av} \quad [79]$$

$$DsRA = [\mathbf{D}_{an}:\mathbf{D}'_{an}/N_D(N_D - 1)D_{av}D'_{av}]^{1/2} \quad [80]$$

$$DFA = (N_D\mathbf{D}_{an}:\mathbf{D}'_{an}/(N_D - 1)\mathbf{D}:\mathbf{D}')^{1/2} \quad [81]$$

$$LI_N = \sqrt{\frac{N_D}{2(N_D-1)^2}}\sqrt{\frac{\mathbf{D}_{an}:\mathbf{D}'_{an}}{\mathbf{D}:\mathbf{D}'}} + \frac{N_D}{2(N_D-1)}\frac{\mathbf{D}_{an}:\mathbf{D}'_{an}}{\sqrt{\mathbf{D}:\mathbf{D}}\sqrt{\mathbf{D}':\mathbf{D}'}} \quad [82]$$

$$dVF/dA = N_D(N_D-1)A[1-A(N_D-2)] \quad [83]$$

Contrast-to-Noise Ratios of Diffusion Anisotropy Indices

In MRI, the contrast-to-noise ratio (*CNR*) typically is interpreted as indicating the ability to differentiate two tissues or regions of interest (ROIs) visually. When applied to DAIs and DAI maps, *CNR* also refers to the ability to tell whether two different ROIs—different tissues within one brain or one group of brains, or the same tissue in two different brains or groups of brains—have the same anisotropy level or statistically different anisotropy levels. It is important to remember this distinction between a visual *CNR* and a statistical *CNR* in the following discussion.

The *CNR* can be defined at a single anisotropy value, or between two very different anisotropy values (*22*). At a single anisotropy value, consider the DAI curves in Fig. 1. The relative contrast at each point is given by

$$\text{contrast} = dDAI/dA = dDAI/dsRA \quad [84]$$

For this definition of contrast, the derivative could be taken with respect to any DAI. *A* or *sRA* was chosen as the reference DAI because for cylindrically symmetric diffusion, *A* and *sRA* vary linearly with the largest eigenvalue (*5*, *21*). When the *CNR*s of two DAIs are being compared, the choice of reference DAI will not affect the *CNR* ratio (see Eqs. [86–88]).

The relative noise at a given anisotropy value for one DAI compared with another DAI depends on whether the two DAIs have a definite algebraic relationship, as in Eqs. [5–11]. If one of these equations applies, then propagation-of-error theory (which is discussed in greater detail in part III) predicts that the ratio of the variances of two DAIs equals the square of the derivative of one DAI with respect to the other. For example,

$$(\sigma_{FA}/\sigma_{sRA})^2 = (dFA/dsRA)^2 \quad [85]$$

Thus, for two DAIs such as *FA* and *sRA*,

$$CNR_{FA} = (dFA/dA)/\sigma_{FA} \quad [86]$$

$$CNR_{sRA} = (dsRA/dA)/\sigma_{sRA} \quad [87]$$

$$CNR_{FA}/CNR_{sRA} = (dFA/dsRA)/(\sigma_{FA}/\sigma_{sRA}) = 1 \quad [88]$$

Notice that the use of *A* as a convenient reference for the derivatives does not affect the ratio CNR_{FA}/CNR_{sRA}

Equation [88] shows that any pair of DAIs related algebraically, as in Eqs. [5–11] and [42–45], should have the same *CNR* for small anisotropy differences. Of course, propagation-of-error theory assumes a first-order linear approximation of the function, and this assumption becomes worse as noise increases. Even if two DAIs are not related algebraically, over a small anisotropy interval there is some algebraic relationship that provides a good approximation of one DAI in terms of the other DAI. Therefore, for low noise levels, all DAIs should have the same *CNR* for very small anisotropy differences. These predictions have been validated in simulations (*22*).

As the difference between the two anisotropy levels increases, the noise term in *CNR* must consider that the noise generally will be different at each anisotropy level. The general *CNR* formula is

$$CNR_2 = \frac{DAI_2 - DAI_1}{(A_2 - A_1)\sqrt{\sigma_1^2 + \sigma_2^2}} \quad [89]$$

where DAI_1 and DAI_2 are the expectation values (averages) of the DAI calculated at each *A* value. The factor $(A_2 - A_1)$ in the denominator is simply a scale factor that allows comparison of *CNR* over different anisotropy intervals. It can be ignored when two DAIs are being compared over the same anisotropy interval.

When two *CNR*s such as CNR_{FA} and CNR_{sRA} are being compared, application of Eq. [89] to each index yields

$$\frac{CNR_{FA}}{CNR_{sRA}} = \frac{FA_2 - FA_1}{\sqrt{(\sigma_{FA1}^2 + \sigma_{FA2}^2)}}\frac{\sqrt{(\sigma_{sRA1}^2 + \sigma_{sRA2}^2)}}{sRA_2 - sRA_1} \quad [90]$$

Dividing both numerator and denominator by σ_{sRA1} yields

$$\frac{CNR_{FA}}{CNR_{sRA}} = \frac{FA_2 - FA_1}{\left(\frac{\sigma_{FA1}}{\sigma_{sRA1}}\right)\sqrt{1 + \left(\frac{\sigma_{FA2}}{\sigma_{FA1}}\right)^2}}\frac{\sqrt{1 + \left(\frac{\sigma_{sRA2}}{\sigma_{sRA1}}\right)^2}}{sRA_2 - sRA_1} \quad [91]$$

which can be rearranged with Eq. [85] to

$$\frac{CNR_{FA}}{CNR_{sRA}} = \frac{FA_2 - FA_1}{\left(\frac{dFA_1}{dsRA_1}\right)\sqrt{\left[1 + \left(\frac{\sigma_{sRA2}}{\sigma_{sRA1}}\right)^2 \frac{\left(\frac{dFA_2}{dsRA_2}\right)^2}{\left(\frac{dFA_1}{dsRA_1}\right)^2}\right]}} \frac{\sqrt{\left[1 + \left(\frac{\sigma_{sRA2}}{\sigma_{sRA1}}\right)^2\right]}}{sRA_2 - sRA_1} \quad [92]$$

Similar equations can be written for the other DAI pairs in Eqs. [36–39]. Equation [92] shows that the ratio CNR_{FA}/CNR_{sRA} depends on only three parameters: (*1*) sRA_1, which determines FA_1 (Eq. [5]) and $dFA_1/dsRA_1$ (Eq. [36]); (*2*) sRA_2, which determines FA_2 (Eq. [5]) and $dFA_2/dsRA_2$ (Eq. [36]), and (*3*) the ratio $\sigma_{sRA2}/\sigma_{sRA1}$.

When the two points approach a single value (a very small anisotropy interval), sRA_1 approaches sRA_2, FA_1 approaches FA_2, and σ_{sRA1} approaches σ_{sRA2}. As a result, CNR_{FA}/CNR_{sRA} approaches a value of 1, as in Eq. [88]. In general, the ratio CNR_{FA}/CNR_{sRA} may be greater than 1 or less than 1, depending on the anisotropy interval and the variance in the data at each point. Notice that CNR_{FA}/CNR_{sRA} does not depend on the actual noise levels (only on the ratio $\sigma_{sRA2}/\sigma_{sRA1}$) or on the actual eigenvalues. Of course, noise may introduce a positive bias in the calculated values of *sRA* and *FA*, and this indirectly affects the observed contrast for *sRA* and *FA*.

Once the two anisotropy levels have been chosen, CNR_{FA}/CNR_{sRA} increases when $\sigma_{sRA2}/\sigma_{sRA1}$ increases, and CNR_{FA}/CNR_{sRA} decreases when $\sigma_{sRA2}/\sigma_{sRA1}$ decreases. To understand why this happens, consider that everything outside the square roots in Eq. [92] is fixed by the choice of anisotropy points, and thus is independent of the noise. The terms in the square roots are identical except for the scale factor $(dFA_2/dsRA_2)^2/(dFA_1/dsRA_1)^2$ in the denominator. Because *dFA*/*dsRA* decreases as anisotropy increases (see Figs. 1 and 2), this scale factor is always less than 1. As $\sigma_{sRA2}/\sigma_{sRA1}$ increases, the numerator increases faster than the denominator, so CNR_{FA}/CNR_{sRA} increases. As $\sigma_{sRA2}/\sigma_{sRA1}$ decreases, the numerator decreases faster than the denominator, so CNR_{FA}/CNR_{sRA} decreases. As a consequence of this, if $\sigma_{sRA2}/\sigma_{sRA1}$ increases because of noise-induced bias or biological heterogeneity, then CNR_{FA}/CNR_{sRA} would be correct for that specific situation, but may not reflect the intrinsic properties of each DAI (*22*). A similar analysis could be performed for each other DAI pair in Eqs. [36–39]. In general, if the variance increases where one DAI has a *lower* slope, then that DAI will have a more favorable *CNR*. This does not guarantee that the ratio will be greater than one, just that it will be greater than if the variance increases where that DAI has a *higher* slope.

DAIs have sometimes been compared on the basis of a so-called signal-to-noise ratio (*SNR*) (*5*, *7*, *26*). In this case *SNR* is more properly called "value-to-noise ratio" or *VNR*. Unlike a conventional MR image, where the noise in each voxel is the same as in all the other voxels, noise in parameter maps such as apparent diffusion coefficient (*ADC*) maps and DAI maps varies from voxel to voxel. Thus, the relationships between *SNR* and *CNR* in an MR image do not necessarily apply to *VNR* and *CNR* in DAI maps. In general, *CNR* is more important than *SNR* or *VNR*. For example, *SNR* is highest in proton-density-weighted images, but most pathologies are seen better in T_2-weighted images and their variations, including FLAIR (fluid-attenuated inversion recovery) and DW images. Because the CNR_2 formula (Eq. [89]) is similar to the Student t-test, *CNR* indicates the ability to differentiate tissues with different anisotropy levels.

Furthermore, propagation-of-error theory predicts that a DAI with a higher *VNR* also has a smaller fractional difference between two regions. According to propagation-of-error theory, *VNR* is the value of a DAI divided by its slope, *DAI*/(*dDAI*/*dA*). The fractional difference of a DAI, in the limit of a very small anisotropy change, is the slope (derivative) divided by the value, or (*dDAI*/*dA*)/*DAI*. This is simply the reciprocal of *VNR*. Neither *VNR* nor fractional difference is statistically meaningful. Only *CNR* provides information about the ability to tell whether two brain regions have the same amount of anisotropy or different amounts of anisotropy.

Although *sRA* and *FA* have similar statistical power, *sRA* is a more logical choice because, with cylindrical symmetry, (*1*) evenly spaced *sRA* values produce evenly spaced λ_1 values (and vice versa), and

(*2*) expressing λ_1 in terms of *sRA*, or *sRA* in terms of λ_1, is easy and straightforward. These are not true of *FA*.

Other Measurements of Anisotropy

Although the DAIs discussed above are useful in many applications, sometimes more information about the anisotropy is desired. The linear, planar, and spherical anisotropy measures (*CL*, *CP*, and *CS*) were defined in terms of eigenvalues (*2*, *27*), and later were expressed in terms of *RA* and other rotational invariants (*28*). The *CA* anisotropy index, which is defined in terms of these measures, ranges from 0 to 1.

$$CL = (\lambda_1 - \lambda_2)/I_1 = (\lambda_1 - \lambda_2)/3D_{av} \quad [93]$$

$$CP = 2(\lambda_2 - \lambda_3)/I_1 = 2(\lambda_2 - \lambda_3)/3D_{av} \quad [94]$$

$$CS = \lambda_3/D_{av} \quad [95]$$

$$CL + CP + CS = 1 \quad [96]$$

$$CA = CL + CP = 1 - CS \quad [97]$$

For cylindrical symmetry, $CA = sRA$ for prolate ellipsoids ($\lambda_2 = \lambda_3$) and $CA = 2sRA$ for oblate ellipsoids ($\lambda_1 = \lambda_2$).

Important Points in "Diffusion Anisotropy Indices"

Of the several intravoxel DAIs that have been proposed, *sRA* and *FA* have been used most widely. The common DAIs range from 0 to 1 and do not require tensor diagonalization or eigenvalue sorting (see Tables 1–4). Some DAIs can be expressed algebraically in terms of other DAIs (e.g., *FA* and *sRA*) and in terms of the parameter *A*, which describes a cylindrically symmetric diffusion tensor. Some intervoxel DAIs have also been defined, including the lattice index (Eq. [66]) and the directional-correlation weighted *sRA* and *FA* (Eqs. [60–65]). The common DAIs have the same theoretical *CNR* at a single point (Eq. [88]), and therefore similar statistical power. However, *sRA* is a more logical choice than *FA* because *sRA* is a more linear function of eigenvalues than *FA* (see Fig. 1). The *CNR* between regions with very different anisotropies depends on the relative noise level at each anisotropy value, $\sigma_{sRA2}/\sigma_{sRA1}$, but not on the actual noise level (Eq. [92]). Absolute anisotropy, *AA*, can be calculated even if no $b = 0$ data are available, and can help determine if a change in *sRA* is due to a change in anisotropy (*AA*) or in D_{av}.

MEASURING APPARENT DIFFUSION COEFFICIENTS

The purposes of this section are (*1*) to explain how diffusion is measured with a spin-echo pulse sequence, (*2*) to show how to calculate the DW factor (*b* factor), (*3*) to explain how to minimize or eliminate interference from nondiffusion gradients, and (*4*) to derive a quantitative estimate of the minimum echo time (*TE*) necessary to achieve a given *b* factor.

How Diffusion Is Measured

Diffusion usually is measured with a spin-echo pulse sequence (*16*, *29–31*), though a stimulated-echo pulse sequence is sometimes used to increase the signal intensity available with long diffusion times (*31*). In the vector model of an NMR measurement, spins with the same Larmor (resonance) frequency (isochromats) are represented by a single magnetization vector. The initial magnetization vectors along the *z* axis are rotated into the transverse (*xy*) plane by a radio frequency excitation pulse. The magnetization vectors then precess in the *xy* plane at their individual Larmor frequencies. The net magnetization vector decreases in magnitude because spins in different environments (different isochromats) experience slightly different magnetic fields and therefore have different Larmor frequencies. This process is called dephasing. After a 180° refocusing pulse is applied, the vectors continue precessing, but now they are realigning, or rephasing. If the Larmor frequencies have not changed, the vectors will realign perfectly. The refocused transverse magnetization will be less than the initial transverse magnetization because of spin-spin (T_2) relaxation. If the Larmor frequencies have changed, then realignment will be imperfect, and the refocused signal intensity will decrease.

In a diffusion measurement, the Larmor frequencies are made spatially dependent by applying a magnetic field gradient in one direction (Fig. 5). For example, if the gradient is applied in the *x* direction, then the Larmor frequency varies with position in the *x* direction. An identical gradient is placed on each side of the 180° refocusing pulse, so spins that have not moved will realign at the echo time. This use of pulsed field gradients in a spin-echo sequence was introduced by Stejskal and Tanner (*32*, *33*) and therefore often is called a Stejskal-Tanner pulse sequence. If spins diffuse randomly, their precession frequency

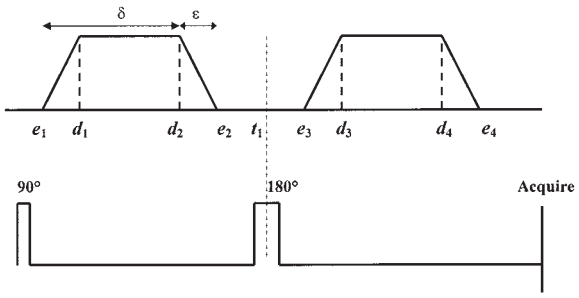

Figure 5 Diagram of diffusion-sensitizing gradients (top line) and their relationship to the radio frequency pulses of a spin-echo sequence (bottom line). The two gradient pulses usually are not symmetrically distributed around the 180° refocusing pulse, which occurs between the two gradient pulses. This pulse sequence is known as a Stejskal-Tanner pulse sequence.

after the refocusing pulse will be different from the frequency before the pulse, and they will not realign perfectly. The more that spins diffuse in the x direction during the spin echo period (TE), the more signal intensity is lost. Spins that move more slowly in the x direction retain more of their net signal intensity. The amount of signal loss depends on the strength and duration of the magnetic field gradient (represented by the letter b) and on D. Signal intensity (SI) in homogeneous, isotropic systems such as a pure liquid decreases exponentially as b increases:

$$SI = S_0'\exp[-TE(b)/T_2]\exp(-bD) = S_0\exp(-bD) \quad [98]$$

where S_0, the signal intensity without diffusion-sensitizing gradients, includes contributions from proton density and T_2 relaxation, whereas S_0' does not include T_2 effects (*30*). The effect of the b factor on the minimum possible TE is discussed later. In many biological systems, signal decay appears to be biexponential (*34–37*), but monoexponential decay is a reasonably good model for b factors up to about 2,000 s/mm^2.

The ADC can be calculated by making measurements at a low b factor, b_1, and a higher b factor, b_2, with all other pulse sequence parameters identical.

$$S_1 = S_0\exp(-b_1D) \quad [99]$$

$$S_2 = S_0\exp(-b_2D) \quad [100]$$

$$S_1/S_2 = \exp[(b_2 - b_1)D] \quad [101]$$

$$D = \ln(S_1/S_2)/(b_2 - b_1) \quad [102]$$

In many cases it is safe to assume that $b_1 = 0$, because the imaging gradients usually provide a very low b factor. The same factor will appear in both the b_1 and b_2 acquisitions, so it will disappear in the term $b_2 - b_1$. In practice it is sometimes necessary to consider interactions between the diffusion gradients and the imaging gradients.

Calculation of *b* Factors with a Single Gradient

A typical set of diffusion-sensitizing gradients is shown in Fig. 5, along with the radio frequency pulses that form the spin echo. The maximum gradient strength is G, the rise and fall times are

$$\varepsilon = d_1 - e_1 = e_2 - d_2 = d_3 - e_3 = e_4 - d_4 \quad [103]$$

the gradient duration is

$$\delta = d_2 - e_1 = e_2 - d_1 = d_4 - e_3 = e_4 - d_3 \quad [104]$$

and the gradient separation time is

$$\Delta = e_3 - e_1 = d_3 - d_1 = d_4 - d_2 = e_4 - e_2 \quad [105]$$

For now, we ignore the imaging gradients and assume that the diffusion-sensitizing gradient is applied in a single direction. Later, we see what happens when two or more gradients are applied together, and when imaging gradients are present. The b factor is calculated in three steps.

Step 1. Determine the gradient intensity, $g(t'')$, at each time t''.

Step 2. Integrate $g(t'')$ over time t'' to calculate the total field offset, $F(t')$, and the k-space factor, $k(t') = \gamma F(t')$, at each time t':

$$k(t') = \gamma F(t') = \gamma \int_0^{t'} g(t'')dt'' - 2H(t' - t_1)k(t_1^-) \quad [106]$$

where

$$k(t_1^-) = \gamma F(t_1^-) = \gamma \int_0^{t_1} g(t'')dt'' \quad [107]$$

The Heaviside unit-step function,

$$H(t) = 0 \quad \text{if } t < 0, \qquad H(t) = 1 \quad \text{if } t \geq 0 \quad [108]$$

accounts for the fact that the 180° refocusing pulse at $t = t_1$ inverts the accumulated k-space factor at that point.

Step 3. Integrate $k^2(t')$ over time t' to calculate the b factor at each time t. The b factor at $t = TE$ is the overall b factor.

$$b(t) = \int_0^t k^2(t')dt' = \gamma^2 \int_0^t F^2(t')dt' \quad [109]$$

$$b(t) = \gamma^2 \int_0^t \left[\int_0^{t'} g(t'')dt'' - 2H(t' - t_1)k(t_1^-) \right]^2 dt' \quad [110]$$

These equations incorporate the refocusing pulse correction (the Heaviside unit-step function) into k (or F), so that k is the k-space factor. Most published equations appear to consider k (or F) to continue increasing monotonically, so that after the refocusing pulse k no longer equals the k-space factor. That approach then corrects for the refocusing pulse by including the Heaviside function when calculating b from F (*33*, *38*, *39*) or k (*40*).

To see how the Heaviside function works, consider k before and after the 180° refocusing pulse. Before the refocusing pulse,

$$k(t' < t_1) = \gamma F(t' < t_1) = \gamma \int_0^{t'} g(t'')dt'' \quad [111]$$

After the refocusing pulse, recalling Eq. [107],

$$k(t_1^+) = \gamma F(t_1^+) = -k(t_1^-) = k(t_1^-) - 2k(t_1^-) \quad [112]$$

$$k(t_1^+) = -\gamma \int_0^{t_1} g(t'')dt'' = \gamma \int_0^{t_1} g(t'')dt'' - 2k(t_1^-) \quad [113]$$

$$k(t' > t_1) = \gamma F(t' > t_1) = \gamma \int_0^{t'} g(t'')dt'' - 2k(t_1^-) \quad [114]$$

Combining Eqs. [108], [111], and [114] yields Eq. [106].

The values of g and F at every time point of the gradient scheme shown in Fig. 5, and of the integral of F^2dt over each interval, are shown in Table 5. For the trapezoidal gradient scheme shown in Fig. 5,

$$b = \gamma^2 G^2[\delta^2(\Delta - \delta/3) + \varepsilon^2/30 - \delta\varepsilon^2/6] \quad [115]$$

If the ramp-up and ramp-down times are ignored ($\varepsilon = 0$), the result for rectangular gradients is

$$b = \gamma^2 G^2 \delta^2 (\Delta - \delta/3) \quad [116]$$

At the other extreme, if the plateau period is eliminated so that $\varepsilon = \delta$,

$$b = \gamma^2 G^2 \delta^2 (\Delta - 7\delta/15) \quad [117]$$

Notice that using a negative diffusion gradient amplitude (a gradient with the same magnitude but in the exact opposite direction) has no effect on the b factor, because the gradient integral always appears squared, as in Eq. [110].

Calculation of *b* Factors with Multiple Gradients

The preceding calculation provides the b factor for a gradient in a single direction. In DTI it is necessary to apply gradients in more than one direction simultaneously. The gradient directions must be specified in three orthogonal directions. In this case the b factor is replaced by a **b** matrix, 3 × 3 for 3D and 2 × 2 for 2D. The orthogonal directions are usually taken to be the x, y, and z axes of the magnet's gradient system, but they also could be the slice-select, frequency-encoding, and phase-encoding directions of each image. For the x, y, and z axes, the 3D **b** matrix consists of six distinct terms: b_{xx}, b_{yy}, b_{zz}, $b_{xy} = b_{yx}$, $b_{xz} = b_{zx}$, and $b_{yz} = b_{zy}$. The three steps are modified as follows:

Step 1. Determine the gradient intensity in each direction at each time t'' to determine the gradient column vector

$$\mathbf{G}(t'') = [g_x(t''), g_y(t''), g_z(t'')]^{\mathrm{T}} \quad [118]$$

Step 2. Integrate $\mathbf{G}(t'')$ over time t'' to calculate the total field offset column vector

$$\mathbf{F}(t') = [F_x(t'), F_y(t'), F_z(t')]^{\mathrm{T}} \quad [119]$$

Table 5 Calculation of *g*, *F*, and *b* for the Gradient Pulse Sequence in Fig. 5

t	g	F	$\int F^2$
$t \le e_1$	0	0	0
$e_1 < t < d_1$	$G(t - e_1)/\varepsilon$	$G(t - e_1)^2/2\varepsilon$	$G^2\varepsilon^3/20$
d_1	G	$G\varepsilon/2$	
$d_1 < t < d_2$	G	$G(\varepsilon/2 + t - d_1)$	$(G^2/3)(\delta^3 - 3\delta^2\varepsilon/2 + 3\delta\varepsilon^2/4 - \varepsilon^3/4)$
d_2	G	$G(\delta - \varepsilon/2)$	
$d_2 < t < e_2$	$G(e_2 - t)/\varepsilon$	$G[\delta - (e_2 - t)^2/2\varepsilon]$	$G^2[\delta^2\varepsilon - \delta\varepsilon^2/3 + \varepsilon^3/20]$
e_2	0	$G\delta$	
$e_2 < t < t_1$	0	$G\delta$	$G^2\delta^2(t_1 - e_2)$
t_1^-	0	$G\delta$	
t_1^+	0	$-G\delta$	
$t_1 < t < e_3$	0	$-G\delta$	$G^2\delta^2(e_3 - t_1)$
e_3	0	$-G\delta$	
$e_3 < t < d_3$	$G(t - e_3)/\varepsilon$	$-G[\delta - (t - e_3)^2/2\varepsilon]$	$G^2[\delta^2\varepsilon - \delta\varepsilon^2/3 + \varepsilon^3/20]$
d_3	G	$-G(\delta - \varepsilon/2)$	
$d_3 < t < d_4$	G	$-G(\delta - \varepsilon/2 - t + d_3)$	$(G^2/3)(\delta^3 - 3\delta^2\varepsilon/2 + 3\delta\varepsilon^2/4 - \varepsilon^3/4)$
d_4	G	$-G\varepsilon/2$	
$d_4 < t < e_4$	$G(e_4 - t)/\varepsilon$	$-G(e_4 - t)^2/2\varepsilon$	$G^2\varepsilon^3/20$
$t \ge e_4$	0	0	

Note: The last column is the integral of F^2 over the time interval in the first column. The sum of the entries in the last column is $G^2[\delta^2(\Delta - \delta/3) + \varepsilon^3/30 - \delta\varepsilon^2/6]$.

and the *k*-space column vector

$$\mathbf{k}(t') = \gamma\mathbf{F}(t') = [k_x(t'), k_y(t'), k_z(t')]^{\mathrm{T}} = [\gamma F_x(t'), \gamma F_y(t'), \gamma F_z(t')]^{\mathrm{T}} \quad [120]$$

at each time t':

$$\mathbf{k}(t_1^-) = \gamma\mathbf{F}(t_1^-) = \gamma\int_0^{t_1}\mathbf{G}(t'')dt'' \quad [121]$$

$$\mathbf{k}(t') = \gamma\mathbf{F}(t') = \gamma\int_0^{t'}\mathbf{G}(t'')dt'' - 2H(t' - t_1)\mathbf{k}(t_1^-) \quad [122]$$

For example,

$$k_x(t_1^-) = \gamma F_x(t_1^-) = \gamma\int_0^{t_1} g_x(t'')dt'' \quad [123]$$

$$k_x(t') = \gamma F_x(t') = \gamma\int_0^{t'} g_x(t'')dt'' - 2H(t' - t_1)k_x(t_1^-) \quad [124]$$

Step 3. Integrate k_ik_j over time t to calculate the b_{ij} matrix element. That is, integrate ${k_x}^2$ to calculate b_{xx}, k_xk_y to calculate b_{xy}, and so on. The **b** matrix elements at $t = TE$ are the overall **b** matrix elements. For example,

$$b_{yz}(t) = \int_0^t k_y(t')k_z(t')dt' = \gamma^2\int_0^t F_y(t')\,F_z(t')dt' \quad [125]$$

$$b_{yz}(t) = \gamma^2\int_0^t\left[\int_0^{t'} g_y(t'')dt'' - 2H(t' - t_1)k_y(t_1^-)\right] \times\left[\int_0^{t'} g_z(t'')dt'' - 2H(t' - t_1)k_z(t_1^-)\right]dt' \quad [126]$$

The entire **b** matrix can be expressed more concisely in terms of the vectors **G** and **k** (*41*):

$$\mathbf{b}(t) = \int_0^t\mathbf{k}(t')\mathbf{k}^{\mathrm{T}}(t')dt' = \gamma^2\int_0^t\mathbf{F}(t')\mathbf{F}^{\mathrm{T}}(t')dt' \quad [127]$$

$$\mathbf{b}(t) = \gamma^2 \int_0^t \left[\int_0^{t'} \mathbf{G}(t'')dt'' - 2H(t' - t_1)\mathbf{k}(t_1^-) \right] \times \left[\int_0^{t'} \mathbf{G}(t'')dt'' - 2H(t' - t_1)\mathbf{k}(t_1^-) \right]^{\mathrm{T}} dt' \quad [128]$$

Notice that the first part of the integrand is a column vector, whereas the second part is a row vector, so that vector multiplication produces the nine terms of the 3×3 **b** matrix, not just one term.

To calculate the signal intensity, the exponent Bd in Eq. [98] is replaced by the generalized dot product between the **b** matrix and the diffusion tensor **D**:

$$\mathbf{b} : \mathbf{D} = \int_0^{TE} \mathbf{k}^{\mathrm{T}}(t')\mathbf{D}\mathbf{k}(t')dt' = \gamma^2 \int_0^{TE} \mathbf{F}^{\mathrm{T}}(t')\mathbf{D}\mathbf{F}(t')dt' \quad [129]$$

$$\mathbf{b}{:}\mathbf{D} = b_{xx}D_{xx} + b_{yy}D_{yy} + b_{zz}D_{zz} + 2b_{xy}D_{xy} + 2b_{xz}D_{xz} + 2b_{yz}D_{yz} \quad [130]$$

$$SI = S_0 \exp(-\mathbf{b}{:}\mathbf{D}) \quad [131]$$

Notice that, in contrast to Eq. [127], the row vector $\mathbf{k}^{\mathrm{T}}$ is on the left and the column vector **k** is on the right in Eq. [129].

Calculation of the **b** matrix for each gradient direction in a DTI measurement is simplified by two common practices. First, the diffusion gradients are applied so that the total gradient amplitude is the same in each individual DWI acquisition. Thus, the overall gradient vector can be expressed as the product of a gradient magnitude, G, and the normalized gradient direction vector $\mathbf{G}_{\mathrm{n}}$, so that

$$\mathbf{G}_{\mathrm{n}} = (g_x, g_y, g_z)^{\mathrm{T}} \quad [132]$$

$$g_x^2 + g_y^2 + g_z^2 = 1 \quad [133]$$

Therefore, each individual acquisition has the same b factor. Second, in each individual DWI acquisition the gradients are always applied with a constant ratio; for example $g_x(t'') = -g_y(t'')$ or $g_y(t'') = 2g_z(t'')$. Therefore, if the imaging gradients can be ignored, the relative magnitudes of the b_{ij} terms can be calculated from the product of the normalized gradient magnitudes, $g_i g_j$.

Table 6 Calculation of the b Matrix for a Common Six-Direction DTI Gradient-Encoding Scheme

Relative **b** Matrix Elements								
g_x	g_y	g_z	b_{xx}	b_{yy}	b_{zz}	b_{xy}	b_{xz}	b_{yz}
1	u	0	1	u^2	0	u	0	0
1	$-u$	0	1	u^2	0	$-u$	0	0
0	1	u	0	1	u^2	0	0	u
0	1	$-u$	0	1	u^2	0	0	$-u$
u	0	1	u^2	0	1	0	u	0
$-u$	0	1	u^2	0	1	0	$-u$	0

Note: The components g_x, g_y, and g_z are in units of the reciprocal of the total gradient magnitude $(1 + u^2)^{1/2}$, so that the actual component magnitudes are $1/(1 + u^2)^{1/2}$ and $u/(1 + u^2)^{1/2}$.

$$\mathbf{g} = \mathbf{G}_{\mathrm{n}}\mathbf{G}_{\mathrm{n}}^{\mathrm{T}} = \begin{pmatrix} g_x \\ g_y \\ g_z \end{pmatrix} (g_x \quad g_y \quad g_z) = \begin{pmatrix} g_x^2 & g_x g_y & g_x g_z \\ g_y g_x & g_y^2 & g_y g_z \\ g_z g_x & g_z g_y & g_z^2 \end{pmatrix} \quad [134]$$

$$\mathbf{b} = b\mathbf{g} \quad [135]$$

$$\mathbf{b}{:}\mathbf{D} = b\mathbf{g}{:}\mathbf{D} \quad [136]$$

An example is shown in Table 6 for a six-direction gradient sampling scheme. When $u = 1$, this produces six pairs of equal-magnitude gradients. When

$$u = \tau \equiv (\sqrt{5} + 1)/2 = 2\cos(\pi/5) \approx 1.618 \quad [137]$$

or

$$u = \tau \equiv (\sqrt{5} - 1)/2 \approx 0.618 \quad [138]$$

this produces gradients pointing to the vertices of a regular icosahedron (*42*). Selection of gradient directions is discussed in more detail in the section on selecting gradient directions.

Eliminating Effects of the Imaging Gradients and Background Gradients

In general, calculation of the exact **b** matrix for an imaging sequence requires knowledge of all gradients present during the imaging sequence, including static background gradients and the imaging gradients as well as the diffusion gradients. However, the imaging

gradients may not be known until the slice orientation has been selected during the MRI exam, and background gradients have unknown magnitudes that vary among voxels. Because imaging gradients and background gradients have similar effects, the following discussion refers to imaging gradients. Identical considerations apply to background gradients.

In the presence of both diffusion gradients and imaging gradients, there will be three types of gradient terms in the overall **b** matrix: $\mathbf{B}_d$ from the diffusion gradients (e.g., $g_{dx}g_{dy}$); $\mathbf{B}_i$ from the imaging gradients (e.g., $g_{ix}g_{iy}$); and $\mathbf{B}_{di}$ from cross-terms involving both diffusion gradients and imaging gradients (e.g., $g_{dx}g_{ix}$). The $\mathbf{B}_d$ terms are the desired diffusion-measurement terms. The $\mathbf{B}_i$ terms also appear in the $b = 0$ images, and therefore will cancel in the ratio S_1/S_2 (Eq. [101]).

The $\mathbf{B}_{di}$ cross-terms can be treated in several ways. First, if the imaging gradients are small, they may be ignored as a first approximation. The results will not be exact, but may be good enough for many purposes. Second, if each diffusion gradient is refocused before the next imaging gradient is turned on, and each imaging gradient is refocused before the next diffusion gradient comes on, there are no $\mathbf{B}_{di}$ cross-terms (*43*). However, this greatly reduces the *b* factor available in a given echo time *TE*, or greatly increases the *TE* to achieve a desired *b* factor. Third, data in each direction can be acquired twice, once with the original diffusion gradients (g_{dx}, g_{dy}, g_{dz}) and once with the negative of each diffusion gradient ($-g_{dx}$, $-g_{dy}$, $-g_{dz}$)—for example, (0.6, −0.8, 0) and (−0.6, 0.8, 0). This will have no effect on $\mathbf{B}_i$, which has terms such as $g_{ix}g_{iy}$, or on $\mathbf{B}_d$, which has terms such as $g_{dx}g_{dy}$. However, the $\mathbf{B}_{di}$ terms such as $g_{dx}g_{ix}$ will all become negative—for example, $g_{dx}g_{ix}$ becomes $-g_{dx}g_{ix}$. Thus the relative signal intensities will be

$$S_2^+ = S_0\exp[-(\mathbf{B}_d + \mathbf{B}_i + \mathbf{B}_{di}):\mathbf{D}] \quad [139]$$

$$S_2^- = S_0\exp[-(\mathbf{B}_d + \mathbf{B}_i - \mathbf{B}_{di}):\mathbf{D}] \quad [140]$$

$$(S_2^+S_2^-)^{1/2} = S_0\exp[-(\mathbf{B}_d + \mathbf{B}_i):\mathbf{D}] \quad [141]$$

Taking the geometric mean of the two images, one with the original diffusion gradients and one with the negative of the original diffusion gradients, eliminates the $\mathbf{B}_{di}$ cross-terms between diffusion gradients and imaging gradients (*44*, *45*). The remaining $\mathbf{B}_i$ term is present with $b = 0$ and $b > 0$, and therefore will not affect calculation of the *ADC* from Eq. [102].

This method of eliminating the effects of cross-terms between diffusion and imaging gradients requires that data from each pair of opposite directions be combined into a single value (Eq. [141]) before the tensor elements are calculated. If the same data are treated as independent directions, it is reasonable to expect that the cross-terms will be greatly reduced, but not entirely eliminated. However, this has not yet been demonstrated either theoretically (analytically or numerically) or in practice.

It is reasonable to expect that the cross-terms will be minimized if the diffusion gradients are uniformly placed around the entire sphere, not just over one hemisphere, even if the directions are not exact +/− pairs. This has not yet been demonstrated either theoretically (analytically or numerically) or in practice. One way to create such a "balanced" gradient scheme is to minimize the residual unbalanced gradient components (*6*). That is, one can sum the components in the *x*, *y*, and *z* directions, square each sum, add the three squares, and minimize this final sum-of-squares (*46*).

Effect of *b* Factor on Minimum *TE*

The minimum possible *TE* for an imaging sequence, TE_{min}, increases as *b* increases. Although it is impossible to determine a single precise relationship that is valid for all situations, the TE_{min} for a given *b* value, or conversely the maximum *b* factor for a given *TE*, b_{max}, can be calculated for a standard echo planar imaging (EPI) pulse sequence (*47*). The formulas in (*47*) contain some errors, so the derivation of those formulas is repeated here. The pulse scheme shown in Fig. 6 assumes square gradient pulses and a full *k*-space echo-planar image acquisition, so that there is minimal time between the 180° refocusing pulse and the start of the second diffusion gradient. There may be some extra time between the end of the first diffusion gradient and the 180° refocusing pulse. With this pulse sequence, *tRF*1 is the minimum possible time between the middle of the 90° excitation pulse and the start of the first diffusion gradient, *SSLC* is the minimum possible time from the end of the first diffusion gradient to the middle of the 180° refocusing pulse, *SSRC* is the minimum possible time from the middle of the 180° refocusing pulse to the beginning of the second diffusion gradient, and *tEPI* is the minimum possible time from the end of the second diffusion gradient to the peak echo formation. The maximum duration of the diffusion gradient is

$$\delta_{max} = TE/2 - t_A \quad [142]$$

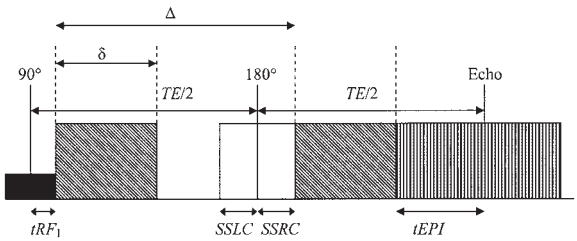

Figure 6 Diagram of a typical spin-echo pulse sequence, showing the two RF pulses (black box and white box), the DW gradients (diagonal hatching), and the echo planar imaging (EPI) acquisition period (vertical stripes). The echo time is *TE*, the DW gradient duration is δ, and the DW gradient separation is Δ. tRF_1 is the minimum time between the 90° excitation RF pulse and the beginning of the first DW gradient, *SSLC* is the minimum time between the end of the first DW gradient and the 180° refocusing RF pulse, *SSRC* is the minimum time between the end of the 180° RF pulse and the beginning of the second DW gradient, and *tEPI* is the minimum time between the end of the second DW gradient and *TE*. Because $SSRC + tEPI > tRF_1 + SSLC$, there is extra time between the first DW gradient and the 180° refocusing rf pulse. This figure is intended to be similar, but not identical, to Fig. 1 in (*47*), with similar labels for the various time periods. To allow clear labels, the relative time periods are not the same as those usually found in MRI pulse sequences.

where

$$t_A = SSRC + tEPI \quad [143]$$

The maximum gradient separation is

$$\Delta_{\max} = TE/2 - t_B \quad [144]$$

where

$$t_B = tRF1 - SSRC \quad [145]$$

Substitution of these values of δ and Δ into Eq. [116] yields

$$b_{\max} = \gamma^2 G^2 (TE/2 - t_A)^2 [(TE/2 - t_B) - (TE/2 - t_A)/3] \quad [146]$$

which can be rearranged to

$$b_{\max} = [\gamma^2 G^2/12][TE^3 - 3(t_A + t_B)TE^2 + 12t_A t_B TE + 4t_A^3 - 12t_A^2 t_B] \quad [147]$$

This is a cubic equation in *TE*,

$$TE^3 - 3(t_A + t_B)TE^2 + 12t_A t_B TE + 4t_A^3 - 12t_A^2 t_B - 12b_{\max}/\gamma^2 G^2 = 0 \quad [148]$$

which can be solved analytically (*47*, *48*) to give

$$TE_{\min} = \kappa + \beta/\kappa + \varepsilon \quad [149]$$

where

$$\alpha = (t_A - t_B)^3 - 6b_{\max}/\gamma^2 G^2 \quad [150]$$

$$\beta = (t_A - t_B)^2 \quad [151]$$

$$\kappa = [-\alpha + (\alpha^2 - \beta^3)^{1/2}]^{1/3} \quad [152]$$

$$\varepsilon = t_A + t_B = tRF1 + tEPI \quad [153]$$

If $\alpha^2 - \beta^3 \geq 0$, κ^3 has one real root and two complex conjugate roots, and $TE_{\min}$ is derived from the real root. If $\alpha^2 - \beta^3 < 0$, then κ^3 has three distinct complex roots. For each root, $\beta/\kappa = \kappa^*$ (the complex conjugate of κ), and $TE_{\min}$ is the largest value derived from these three roots. When this minimum *TE* from Eq. [149] is incorporated into the equation for signal intensity, Eq. [98] becomes

$$SI = S_0' \exp[-TE_{\min}(b)/T_2]\exp(-bD) = S_0 \exp(-bD) \quad [154]$$

If $tRF1 + SSLC > tEPI + SSRC$, there is no extra time between the end of the first diffusion gradient and the 180° refocusing pulse, whereas there may be some extra time between the 180° refocusing pulse and the start of the second diffusion gradient (Fig. 7). This could happen with partial κ-space imaging, for

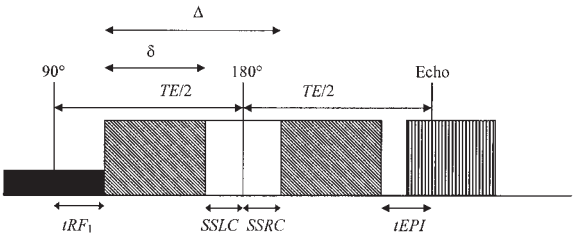

Figure 7 Diagram of a typical spin-echo pulse sequence showing the two RF pulses, the DW gradients (diagonal hatching), and the echo planar imaging (EPI) acquisition period (vertical stripes). In contrast to Fig. 6, there is a longer delay after the 90° pulse and a shorter EPI acquisition period before *TE*. As a result, $tRF_1 + SSLC > SSRC + tEPI$, so there is extra time between the 180° RF pulse and the second DW gradient. The time period labels are the same as in Fig. 6.

example (*49*). In this case, Eqs. [142–145] must be replaced by

$$\delta_{max} = TE/2 - t_C \quad [155]$$

$$t_C = tRF1 + SSLC \quad [156]$$

$$\Delta_{max} = TE/2 - t_D \quad [157]$$

$$t_D = tEPI - SSLC \quad [158]$$

When TE_{min} is calculated in this case, t_A and t_B must be replaced by t_C and t_D in Eqs. [146–153].

The ε term in Eq. [149] represents the required time before and after the diffusion gradients, so it does not change when *b* changes. Therefore, the ε term can be ignored when the signal intensities at two different *b* factors are compared.

For many applications it is not essential to know the exact TE_{min}, only the approximate change in TE_{min} when *b* changes. For such approximate calculations it may be sufficient to ignore $\varepsilon = tRF1 + tEPI$ and to assume that $SSLC = SSRC = 0$ so that $\delta = \Delta = TE/2$ (*50*). In this case, Eq. [116] simplifies to

$$b_{max} = \gamma^2 G^2 (TE/2)^2 [TE/2 - (TE/2)/3] \quad [159]$$

$$b_{max} = \gamma^2 G^2 TE^3/12 \quad [160]$$

which leads to

$$TE_{min} = (12b/\gamma^2 G^2)^{1/3} \quad [161]$$

This expression, much simpler than Eqs. [149–153], can easily be incorporated into expressions for signal intensity as a function of *b*. Equations [149] and [161] are compared in Fig. 8.

For a given *b* factor and G_{max}, TE_{min} can be decreased by using all three gradients instead of one gradient at a time. For example, for a three-direction *ADC* measurement, using $(G_x, G_y, G_z) = (1, 1, -1/2)$, $(1, -1/2, 1,)$, and $(-1/2, 1, 1)$ provides three orthogonal gradient directions with the *b* factor increased by a factor $(1^2 + 1^2 + 0.5^2) = 2.25$ and the effective gradient strength increased by a factor $(1^2 + 1^2 + 0.5^2)^{0.5} = 1.5$.

Important Points in "Measuring Apparent Diffusion Coefficients"

In isotropic liquids, diffusion causes an exponential signal loss based on the product *Bd* (Eq. [98]). Diffusion usually is measured by applying diffusion-sensitizing gradients in the two halves of a spin-echo sequence (before and after the 180° refocusing pulse; see Figs. 5–7). The *b* factor depends on the gradient magnitude and duration, and the delay before the refocusing gradient. Although the imaging gradients can affect the exact *b* factor, an estimate based on the diffusion-sensitizing gradients alone is shown in Eqs. [115–117]. The effect of the imaging gradients can be eliminated by repeating each measurement with the

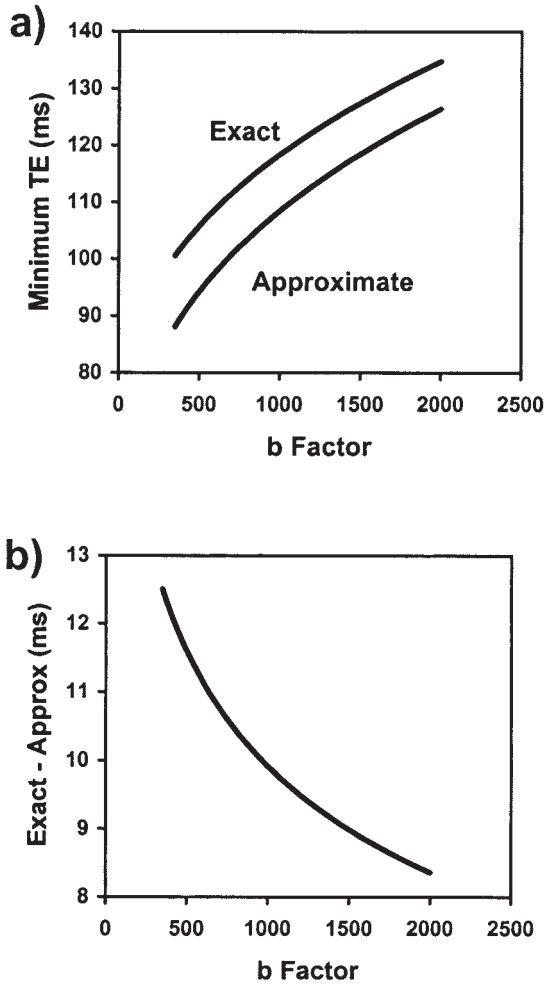

Figure 8 Comparison of the exact formula for TE_{min} as a function of b_{max} (Eq. [149]) and the approximate value from Eq. [161]. Values of *SSRC* and *tEPI*, calculated from the data of (*47*), were added to the estimate of Eq. [161] so that it would be closer to the exact formula. The maximum gradient strength (G_{max}) was assumed to be 22 mT/m. (a) The two calculations seem to undergo similar changes as the *b* factor increases from 350 s/mm^2 to 2000 s/mm^2. (b) A closer look shows that the difference between the two calculations slowly declines from 12.5 ms to 8.4 ms as the *b* factor increases from 350 s/mm^2 to 2000 s/mm^2. The exact TE_{min} increases from 100.5 ms to 134.8 ms, whereas the approximate TE_{min} increases from 88.0 ms to 126.4 ms.

negative gradient direction (Eq. [141]). As the *b* factor increases with a given gradient strength, the minimum echo time TE_{min} also increases (Eq. [149]). An approximate relationship between TE_{min} and the *b* factor is suitable for many applications (Eq. [161]).

SELECTING GRADIENT DIRECTIONS

The purposes of this section are 1) to discuss the importance of choosing a good gradient sampling scheme for DWI or DTI, 2) to introduce the regular polyhedra (Platonic solids), 3) to present some optimal sampling schemes explicitly, 4) to explain how D_{av} and *sRA* sometimes can be calculated without calculating the tensor elements, 5) to discuss some possible ways to evaluate and compare gradient sampling schemes, and 6) to outline some key ideas to consider when selecting a gradient sampling scheme.

General Concerns

Much has been written about the best way to select the directions in which to measure the *ADC* in order to calculate the tensor elements (*5*, *6*, *42*, *47*, *51–54*). If the tensor (diffusion ellipsoid) orientation were known, then three measurements along the three eigenvector directions would provide all the necessary information. Because the tensor orientation generally is not known in advance, and generally many orientations will be present in a biological sample, diffusion must be measured in at least six directions.

Although most of this section considers the optimal gradient sampling schemes, it is sometimes desirable to compare optimal schemes with suboptimal schemes. Some necessary mathematical conditions for gradient sampling schemes that will guarantee a valid tensor calculation have been proposed (*55*). There must be at least one set of six gradient vectors with the following properties (some of these rules can be violated by gradient vectors beyond the chosen six, but at least one set of six vectors must obey all these rules):

1. No two gradient vectors are parallel or antiparallel. That is, no gradient vector can be a (positive or negative) multiple of another vector.
2. If three vectors belong to a 2D subspace (i.e., the vectors are coplanar, so that the three possible cross-products are collinear), then the remaining three vectors must be linearly independent. In other words, the remaining three vectors must not be coplanar.
3. No subset of four gradient vectors may belong to the same 2D subspace. That is, no four vectors may be coplanar.

Adherence to the above three rules does not necessarily guarantee that the tensor can be calculated from the data. However, violation of any of the rules means that an attempt to calculate the tensor will encounter mathematical problems.

There is general agreement that the optimal directions should be spread out as uniformly as possible over a hemisphere, and that some of these directions should then be made negative (pointing toward the other hemisphere) to minimize the effects of cross-terms between the diffusion gradients and the imaging gradients in D_{av}, and presumably in the off-diagonal tensor elements.

There is less agreement about how many directions to use, exactly what directions to use, and which negative directions to use. In general it appears that if the gradient directions are uniformly spaced, then the use of more directions (rather than multiple acquisitions in a few directions) yields more constant variance in the calculated tensor parameters as the tensor is rotated, but no improvement in the average variance over all possible rotations (*50*, *51*).

In the early DTI days, gradient strength was typically limited to about 23 mT/m, so there was a considerable advantage to using two or even three of the *x*, *y*, and *z* gradients for each diffusion direction. This advantage becomes less important as the *TE* decreases with stronger gradients and with parallel imaging techniques. This advantage becomes more important at higher field strengths, such as 3 T or higher, where T_2 values decline.

This article focuses on gradient directions defined by the five regular polyhedra, also called the Platonic solids. These directions can be written concisely with analytic formulas. Suggestions for designing other encoding schemes are also presented. The polyhedra do not specify the polarity of each gradient direction, only its orientation. The gradient schemes given in the tables usually represent one of the most balanced schemes possible, but other equally balanced schemes may be available, and many less-balanced schemes can be derived. The sum-of-squares for each normalized gradient scheme, which was explained at the end of the section on eliminating effects of the imaging gradients and background gradients, is also shown in the tables. The gradient strengths are chosen to be similar to previously published polyhedral gradients whenever possible (*42*).

For each gradient scheme shown below, equivalent schemes can be derived by multiplying all the *x*, *y*, or

Table 7 Regular Polyhedra

Name	Shape	Faces	Edges	Vertices	Directions[a]
Tetrahedron	Triangle (3)	4	6	4	3, 4
Cube	Square	6	12	8	3, 4, 6
Octahedron	Triangle (4)	8	12	6	3, 4, 6
Dodecahedron	Pentagon	12	30	20	6, 10, 15
Icosahedron	Triangle (5)	20	30	12	6, 10, 15

[a] The number of diffusion directions available from the faces, edges, or vertices of each regular polyhedron. Notice that in each case the number of vertices plus the number of faces is two more than the number of edges.

z components by -1. This is equivalent to reflecting the gradients through the yz, xz, or xy plane. The sum-of-squares term remains the same. However, the **b** matrix associated with each gradient direction changes, and the cross-terms between diffusion and imaging gradients may change in a specific situation. If one gradient direction is multiplied by -1, the **b** matrix does not change, but the cross-terms between diffusion and imaging gradients may change.

Descriptions of Regular Polyhedra (Platonic Solids)

A regular polyhedron is a solid whose surface consists of equilateral triangles, squares, or regular pentagons. One solid each is possible with squares or pentagons, and three solids are possible with equilateral triangles, depending on whether three, four, or five triangles meet at each vertex. The five regular polyhedra are summarized in Table 7. In each case the number of vertices plus the number of faces is two more than the number of edges. Several publications have included good pictures of the dodecahedron (*42*) and icosahedron (*6*, *7*, *42*, *46*, *53*), as well as higher levels of uniformly spaced vertices (*42*, *46*). Pictures and further information are available at a number of Internet sites.

The cube and the octahedron are called "duals" of each other because they can be aligned so that each cube face corresponds to one octahedron vertex, and each cube vertex corresponds to one octahedron face (Fig. 9[a]). The dodecahedron and icosahedron form a similar pair. In each of these four polyhedra, a line drawn from one vertex through the center passes through another vertex, a line drawn from one face through the center passes through another face, and a line drawn from one edge through the center passes through another edge. In contrast, a line from one vertex of the tetrahedron through the center passes through the center of a face, and a line from the center of one edge passes through the center of another edge. Therefore, except for the faces and vertices of the tetrahedron, the vertices, edges, or faces of a regular polyhedron define half that number of unique directions. Thus, 3, 4, 6, 10, or 15 uniformly spaced directions can be prescribed from the regular polyhedra. The tetrahedron can be inscribed in a cube so that each vertex and each face corresponds to a vertex of the cube, and each edge corresponds to a face of the cube (see Fig. 9[b]) (*8*).

Certain values recur frequently in defining polyhedral gradient directions, so it is convenient to define these values separately. Icosahedral and dodecahedral directions can be defined by using τ from either Eq. [137] or Eq. [138]. This article uses Eq. [137] to be consistent with previously published encoding schemes (*6*, *42*), though Eq. [138] could also be used (*53*). Use of Eq. [138] would change the orientation of the icosahedron or dodecahedron and would require different formulas in some cases. It is important not to confuse the following parameters with other parameters with similar names: A = anisotropy index (Eqs. [12] and [46]), $D = ADC$, G = gradient vector strength (Eqs. [115–117]), H = Heaviside step function (Eq. [108]), and the rotational invariants proposed by Bahn (*1*), which were discussed in part I.

$$\tau = (\sqrt{5} + 1)/2 = 2\cos(\pi/5) \approx 1.618034 \quad [162]$$

$$A = 1/(1 + \tau^2)^{1/2} = [(5 - \sqrt{5})/10]^{1/2} \approx 0.525731 \quad [163]$$

$$B = \tau/(1 + \tau^2)^{1/2} = [(5 + \sqrt{5})/10]^{1/2} \approx 0.850651 \quad [164]$$

$$C = (\tau - 1)/2 = (\sqrt{5} - 1)/4 \approx 0.309017 \quad [165]$$

$$D = \tau/2 = (\sqrt{5} + 1)/4 \approx 0.809017 \quad [166]$$

$$E = 2^{1/2}/2 \approx 0.707107 \quad [167]$$

$$F = 3^{1/2}/3 \approx 0.577350 \quad [168]$$

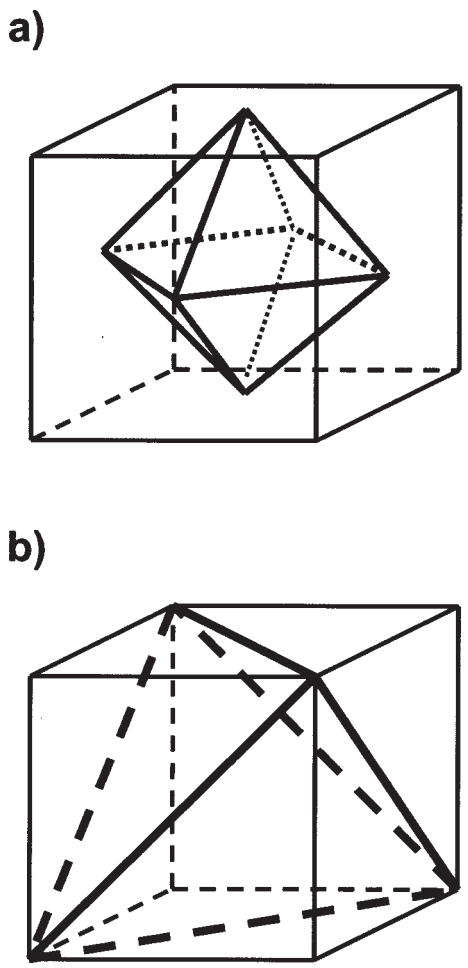

Figure 9 (a) An octahedron inscribed inside a cube. The octahedron comprises eight equilateral triangles, with four triangles meeting at each vertex. The cube and the octahedron are duals of each other: each of the six octahedral vertices corresponds to a cube face, and each of the eight octahedral faces corresponds to a cube vertex (corner). Each of the 12 octahedron edges corresponds to a cube edge. (b) A tetrahedron inscribed in a cube. The tetrahedron comprises four equilateral triangles, with three triangles meeting at each vertex. The cube comprises six squares. Four cube vertices (corners) correspond to tetrahedron vertices. The other four cube vertices correspond to tetrahedron faces. Each of the six tetrahedron edges is along the diagonal of a cube face. Each of the four cube diagonals passes from one tetrahedron vertex, through the center of the cube and tetrahedron, and through the middle of the opposite tetrahedron face. Properties of the tetrahedron, the cube, the octahedron, and the two other regular polyhedra (Platonic solids) are summarized in Table 7.

$$G = (1 + 2\tau)/[3^{1/2}(1 + \tau)] = [(3 + \sqrt{5})/6]^{1/2} \approx 0.934172 \quad [169]$$

$$H = \tau/[3^{1/2}(1 + \tau)] = [(3 - \sqrt{5})/6]^{1/2} \approx 0.356822 \quad [170]$$

$$AB = \sqrt{5}/5 \approx 0.447214 \quad [171]$$

$$CD = 1/4 \quad [172]$$

$$GH = 1/3 \quad [173]$$

$$\tau^2 + 1/\tau^2 = 3 \quad [174]$$

The name of each gradient scheme will include the number of directions and a letter indicating how the directions were chosen (*xyz*, tetrahedron, pairs of gradients, vertices or faces or edges of a polyhedron, or a combined scheme).

Gradient Directions from the Tetrahedron, Cube, and Octahedron

Three orthogonal directions can be specified by the tetrahedron edges, the cube faces, or the octahedron vertices (Table 8). These three directions can be defined as the *x*, *y*, and *z* axes (scheme 3x), or they can be created by selecting a combination of gradients along all three axes (scheme 3a). Scheme 3a can be obtained by rotating scheme 3x by $\pi/6$ radians about the axis (1, 1, 1) (Fig. 10[a]). A variation of scheme 3a has been published (*34*). Compared to scheme 3x, scheme 3a allows a greater *b* factor in a given echo time. In each case, the average of the three measured *ADC*s is D_{av}. The use of one negative direction or a rotated gradient scheme would not decrease the overall cross-term magnitude. Cross-terms can be eliminated by repeating each measurement in the negative direction, for a total of six diffusion-weighted measurements.

Four uniformly spaced directions can be specified by the tetrahedron vertices or faces, the cube vertices, or the octahedron faces. These four directions and their negatives can be defined by the eight vectors $(\pm 1, \pm 1, \pm 1)^T$ (*8*, *56*) and are often called the tetrahedral gradient set (scheme 4t in Table 9). The average of the four measured *ADC*s is D_{av}. If the four directions are chosen so that the sum of the components in each direction (*x*, *y*, and *z*) equals 0, as in scheme 4t in Table 9, then cross-terms are eliminated when D_{av} is calculated (*8*, *56*). Furthermore, because each gradient can be at its maximum strength, the minimum *TE* for a given *b* factor can be achieved. Thus, a cross-term-free measurement of D_{av} with the

Table 8 The *x*, *y*, and *z* Components of Vectors Corresponding to Three Orthogonal Gradient Directions Derived from Regular Polyhedra

	Scheme 3x (sum = 3)			Scheme 3a (sum = 3)		
	1	2	3	1	2	3
Maximum gradient strength						
g_x	1	0	0	−0.5	1	1
g_y	0	1	0	1	−0.5	1
g_z	0	0	1	1	1	−0.5
Normalized gradients						
g_x	1	0	0	−1/3	2/3	2/3
g_y	0	1	0	2/3	−1/3	2/3
g_z	0	0	1	2/3	2/3	−1/3

Note: The sum indicated in parentheses is the sum of squares of the *x*, *y*, and *z* direction sums with normalized gradients. Scheme 7c is 3x + 4t, with set 4t1 negative.

minimum possible *TE* can be achieved with four diffusion-weighted measurements.

The tetrahedral scheme 4t can be combined with scheme 3x to yield a seven-direction scheme 7c (Table 10), which seems to be the optimum seven-direction scheme (*42*, *51*, *57*). This corresponds to all the faces and vertices of the cube or octahedron, or all the vertices and edges of a tetrahedron. The gradients will be more balanced (sum = 0.072 instead of 3) if the gradient set 4t1 is made negative. The *TE* advantage of the tetrahedral scheme 4t is lost when these two schemes are combined, though some of it could be recovered if the entire scheme were rotated appropriately.

The 12 edges of the cube or the octahedron define six gradient-pair directions, from which the entire tensor can be calculated (*2*, *5*, *10*, *42*, *46*, *53*, *54*, *57–59*). An example with maximum gradient amplitudes was shown in Table 6 with $u = 1$, and a normalized and more balanced scheme (sum = 2 instead of six) is shown as scheme 6p in Table 11. This scheme is not as uniformly spread as the icosahedral vertex scheme 6v shown in Table 12 (*42*, *46*), but the use of maximum gradient strengths allows a shorter *TE*.

Images acquired with scheme 6p are shown in Fig. 11. These images are part of the data set from which the DAI maps in Fig. 3 were produced. These images were acquired with a single-shot echo-planar imaging pulse sequence, 5-mm slice thickness, $b = 860$ s/mm^2, $TE = 105$ ms, and a 128×128 matrix over a 24-cm field of view. Notice the dramatic SI difference in the splenium of the corpus callosum, where the intensity is low when the gradient direction is parallel to the nerve fibers and high when the gradient direction is perpendicular to the nerve fibers.

Schemes 3x (cube faces) and 6p (cube edges) could be combined to produce nine directions, scheme 9p. As explained below, this appears to be more uniform than replacing scheme 6p with any other value of u in Table 6, such as scheme 6v, producing scheme 9v (*6*).

Thirteen directions could come from scheme 3x (cube faces) plus scheme 4t (cube corners) plus scheme 6p (cube edges), producing scheme 13e (*42*). The six edges could be replaced by another choice of u in Table 6, such as scheme 6v. The choice of $u = 3^{1/2} - 1 \approx 0.732$, to produce scheme 13o, seems to provide the most uniform spacing, as discussed below.

Gradient Directions from the Icosahedron and Dodecahedron

Normalized and balanced gradients for the icosahedral vertex scheme 6v are shown in Table 12 (*6*, *42*, *46*, *51*, *57*, *59*). The icosahedral scheme 6v provides more uniform spatial coverage than the gradient pair scheme 6p, and it retains some of the *TE* advantage.

The 10 gradient directions for the 20 icosahedral faces can be found by adding or averaging the appropriate triplets of gradients in scheme 6v, followed by normalization, to yield scheme 10f (Table 13) (*42*, *53*). This scheme includes an unbalanced version of scheme 4t, and the other six directions can be created from Table 6 with $u = \tau/(1 + 2\tau)$.

The 15 gradient directions for the 30 icosahedral or dodecahedral edges can be found by adding or averaging the appropriate pairs of gradients in scheme 6v to yield scheme 15e (Table 14) (*6*). This scheme includes a version of scheme 3x.

Schemes 6v (see Table 12), 10f (see Table 13), and 15e (see Table 14) are for an icosahedron (or dodecahedron) with a single orientation. Therefore, they can be combined to make uniformly spaced gradients with 6, 10, 15, 16, 21, 25, or 31 directions (*6*). In these

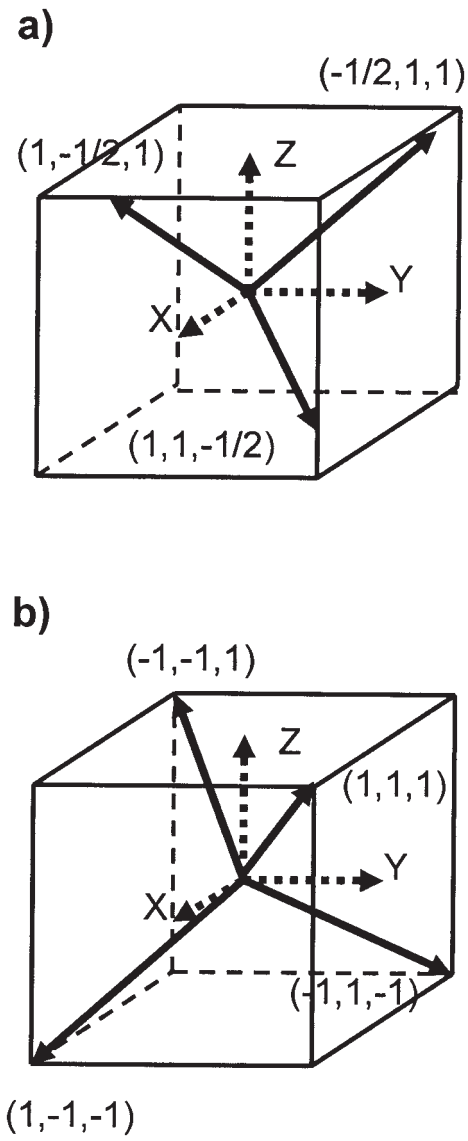

Figure 10 (a) Illustration showing how the three directions corresponding to the *x*, *y*, and *z* axes (Scheme 3x, dotted arrows) can be rotated about the (1, 1, 1) direction to produce three new orthogonal directions (Scheme 3a, solid arrows). (b) Illustration showing how the cube defines three orthogonal directions (dotted arrows), and four uniformly spaced directions (solid arrows). The three orthogonal directions, derived from the six cube faces, correspond to the *x*, *y*, and *z* axes. The four directions are derived from four of the eight cube corners, with no two corners adjacent. Comparison with Fig. 9(b) shows that the four uniformly spaced directions correspond to the tetrahedron vertices.

combined sets, one or more directions may have to be made negative to achieve optimum balance (see Table 10).

All the polyhedral encoding schemes presented here, and the scheme in Table 6 with any value of *u*, allow D_{av} to be calculated as the average of the individual *ADC*s (D_{poly}) without calculation of the tensor elements (*6*, *53*).

$$D_{av} = \overline{D_{poly}} \quad [175]$$

In the absence of noise, each icosahedral scheme also allows calculation of the eigenvalue variance (σ_λ^2) or standard deviation (*SD*) (σ_λ) (*53*), and hence of *sRA* (*6*) or *FA* (*6*, *53*), without calculation of the tensor elements. In the presence of noise, this calculation is correct only for scheme 6v. After calculating the variance of the individual *ADC*s, σ_D^2,

$$\sigma_D^2 = \overline{D_{poly}^2} - \overline{D_{poly}}^2 \quad [176]$$

$$\sigma_\lambda^2 = 2.5\sigma_D^2 \quad [177]$$

$$sRA = \sigma_\lambda/(2^{1/2}D_{av}) = 5^{1/2}\sigma_D/2D_{av} \quad [178]$$

$$FA = \sqrt{\frac{3.75\sigma_D^2}{2.5\sigma_D^2 + D_{av}^2}} \quad [179]$$

The 2D gradient directions corresponding to icosahedral directions are those that uniformly cover the semicircle, so that they are spaced by 180/*M* degrees where *M* is the number of different directions. The resulting 2D formulas are

$$\sigma_\lambda^2 = 2\sigma_D^2 \quad (2D) \quad [180]$$

$$sRA = \sigma_\lambda/D_{av} = 2^{1/2}\sigma_D/D_{av} \quad (2D) \quad [181]$$

Table 9 The *x*, *y*, and *z* Components of Vectors Corresponding to Four Gradient Directions Corresponding to the Four Vertices and Four Faces of a Tetrahedron, or the Eight Vertices of a Cube

Scheme 4t (Sum = 0)				
	1	2	3	4
Maximum gradient strength				
g_x	1	1	−1	−1
g_y	1	−1	1	−1
g_z	1	−1	−1	1
Normalized gradients				
g_x	*F*	*F*	−*F*	−*F*
g_y	*F*	−*F*	*F*	−*F*
g_z	*F*	−*F*	−*F*	*F*

Note: $F = 3^{1/2}/3 \approx 0.577350$ is defined in Eq. [168]. The sum indicated in parentheses is the sum of squares of the *x*, *y*, and *z* direction sums with normalized gradients. Scheme 7c is 3x + 4t, with set 4t1 negative.

Table 10 Summary of Combined Polyhedral-Encoding Schemes

Scheme	From	Sum	Negative Sets	Minimum Sum
7c	3x + 4t	.072	4t1	.072
16c	6v + 10f	1.757	10f5, 6, 8, 10	.147
21c	6v + 15e	2.498	6v5; 15e4, 8–10, 13, 14	.0079
25c	10f + 15e	.959	10f1, 3, 6, 7, 10; 15e3, 6, 11, 12, 14, 15	.00159
31c	6v + 10f + 15e	2.936	6v3, 6; 10f7, 8, 10; 15e1–5, 7, 9, 11, 12, 14	.000457

Note: The individual gradient schemes are shown in Tables 8, 9, 12–14. The sum in the third column is the sum of squares for the combined individual gradient schemes, the minimum sum in the last column is the sum with the negative gradients indicated, and appears to be a minimum sum in each case. Equation [175] applies to all five schemes, and Eqs. [176–178] apply to the last four schemes in the absence of noise.

$$FA = \sqrt{\frac{4\sigma_D^2}{2\sigma_D^2 + D_{\text{av}}^2}} \quad \text{(2D)} \qquad [182]$$

These formulas for *sRA* and *FA* are valid for all integer values of M in the absence of noise, and for $M = 3$ in the presence of noise. Equations [177–182] can be summarized as follows, where N_{D} is the number of dimensions (2 or 3).

$$\sigma_\lambda^2 = [(N_{\text{D}} + 2)/2]\sigma_D^2 \qquad [183]$$

$$sRA = \sigma_\lambda/[(N_{\text{D}} - 1)^{1/2} D_{\text{av}}] = (N_{\text{D}} + 2)^{1/2}\sigma_D/[2(N_{\text{D}} - 1)]^{1/2} D_{\text{av}} \qquad [184]$$

$$FA = \sqrt{\frac{N_{\text{D}}(N_{\text{D}} + 2)\sigma_D^2}{(N_{\text{D}} - 1)(N_{\text{D}} + 2)\sigma_D^2 + 2(N_{\text{D}} - 1) D_{\text{av}}^2}} \qquad [185]$$

Variations of Eqs. [177] and [179] have been published for schemes 6v (*6*, *53*), 10f (*53*), 15e, and 21c (*6*), and Eq. [178] has been published for schemes 6v, 15e, and 21c (*6*). A summary of the properties of the individual polyhedral encoding schemes in Tables 8, 9, and 11–14 is shown in Table 15.

Table 11 The Normalized *x*, *y*, and *z* Gradient Vector Components for Six Gradient Pairs Corresponding to the 12 Cube Edges

	Scheme 6p (Sum = 2)					
	1	2	3	4	5	6
g_x	E	E	0	0	$-E$	$-E$
g_y	E	$-E$	E	E	0	0
g_z	0	0	E	$-E$	$-E$	E

Note: The non-normalized gradient strengths are shown in Table 6 with $u = 1$. $E = 2^{1/2}/2 \approx 0.707107$ is defined in Eq. [167], and set 6p5 has been multiplied by -1 to provide a more "balanced" scheme. The sum indicated in parentheses is the sum of squares of the x, y, and z direction sums with normalized gradients.

Other Gradient Schemes

The optimal choice of other small numbers of directions, such as nine or 13, is not always clear. Many examples can be based on the six directions of Table 6. For example, consider the choice of 10 directions. The most uniform spacing appears to be scheme 10f (see Table 13). As mentioned previously, this is equivalent to scheme 4t plus the six directions in Table 6 with $u = \tau/(1 + 2\tau)$. One could also choose other values of u, as in scheme 6p (producing scheme 10p, four cube corners plus six cube edges) (*42*, *57*) or 6v (producing scheme 10v, six icosahedral vertices plus four of the icosahedral faces). Evaluation of these two schemes, which is described in detail later, confirms that scheme 10f is more uniform than schemes 10p and 10v.

A 12-direction gradient scheme, scheme 12u, can be produced by rotating the six-direction scheme in Table 6 by 90° about the x, y, or z axis. These schemes can be perfectly balanced if either the original six-direction set or the rotated six-direction set is made negative. The choice of $u = 0.5$ appears to provide the

Table 12 The Normalized *x*, *y*, and *z* Gradient Vector Components Corresponding to Six Directions Derived from Icosahedral Vertices or Dodecahedral Faces

	Scheme 6v (Sum = 1.53)					
	1	2	3	4	5	6
g_x	A	A	0	0	B	$-B$
g_y	B	$-B$	$-A$	A	0	0
g_z	0	0	$-B$	$-B$	A	A

Note: The non-normalized gradient strengths are shown in Table 6 with u from Eq. [137], and set 6v3 has been multiplied by -1 to provide a more "balanced" scheme. $A = 1/(1 + \tau^2)^{1/2} = [(5 - 5^{1/2})/10]^{1/2} \approx 0.525731$ is defined in Eq. [163], and $B = \tau/(1 + \tau^2)^{1/2} = [(5 + 5^{1/2})/10]^{1/2} \approx 0.850651$ is defined in Eq. [164]. The sum indicated in parentheses is the sum of squares of the x, y, and z direction sums with normalized gradients.

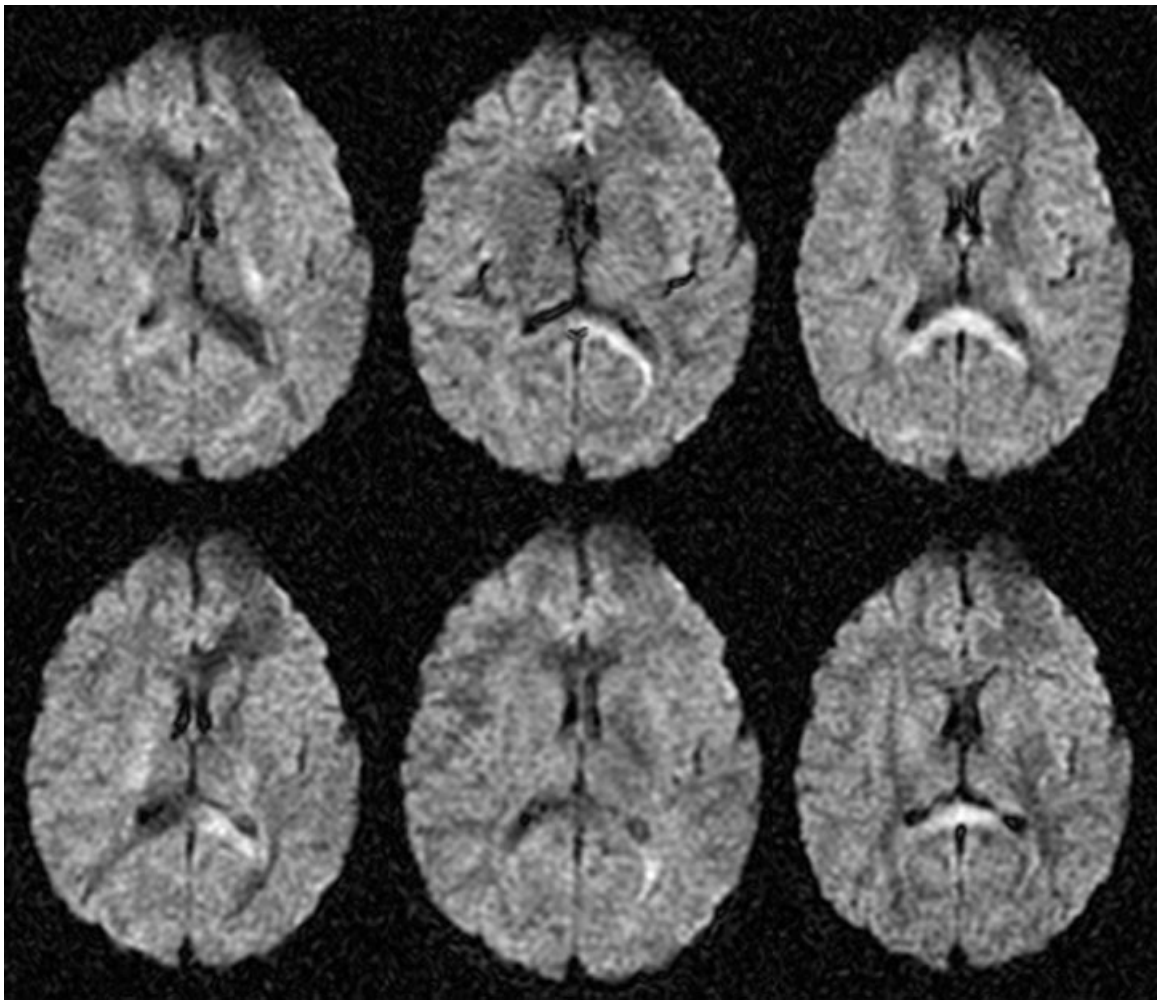

Figure 11 Diffusion-weighted images with six different gradient directions defined by the 12 cube edges (Scheme 6p). Top row: $(g_x, g_y, g_z) = (-1, -1, 0)$, $(1, 0, 1)$, $(0, -1, -1)$. Bottom row: $(g_x, g_y, g_z) = (-1, 1, 0)$, $(-1, 0, 1)$, $(0, -1, 1)$. The axes are defined as $+x$ to the left of the image (subject's right), $+y$ to the top of the image (anterior), and $+z$ into the page (superior). The greatest intensity differences are seen in the splenium of the corpus callosum, where low intensities indicate that the nerve fiber tracts are aligned nearly parallel to the applied diffusion gradient, and high intensities indicate alignment nearly perpendicular to the gradients. These images are part of the raw data set from which the DAI maps in Fig. 3 were produced.

most uniform spacing between gradient directions, as judged from the maximum dot product between any two normalized direction vectors. This scheme has been published, with some typographical errors (*58*). Another way to choose 12 directions is to delete the cube faces (scheme 3x) from the icosahedral edges (scheme 15e) to produce scheme 12e (*6*).

Hasan et al. (*42*) gave references for "face triangulation" of the icosahedron, resulting in $5n^2 + 1$ directions (*42*). This involves joining the midpoints of each edge so that each midpoint becomes a new vertex. Each vector is then normalized to unit magnitude. Scheme 6v corresponds to $n = 1$, and scheme 21c corresponds to $n = 2$. The midpoints of the new edges can then be joined to form 46 directions for $n = 3$, and so on. Other geometric derivatives of the polyhedral encoding schemes are given by Hasan et al. (*42*).

For a large number of encoding directions, N, an analytic encoding scheme can provide a spiral of points that covers the unit sphere with an equal distance between adjacent points (*60*). These formulas were reproduced in (*42*) with a typographical error.

$$g_z(n) = (2n - N - 1)/N \qquad [186]$$

Table 13 The *x*, *y*, and *z* Components of Vectors Corresponding to 10 Gradient Directions Derived from the 20 Icosahedral Faces or Dodecahedral Vertices

	Scheme 10f (Sum = .584)									
	1	2	3	4	5	6	7	8	9	10
g_x	F	$-F$	F	F	$-G$	G	0	0	$-H$	$-H$
g_y	F	F	$-F$	F	$-H$	$-H$	$-G$	G	0	0
g_z	F	F	F	$-F$	0	0	$-H$	H	$-G$	G

Note: $F = 3^{1/2}/3 \approx 0.577350$ is defined in Eq. [168], $G = (1 + 2\tau)/[3^{1/2}(1 + \tau)] = [(3 + 5^{1/2})/6]^{1/2} \approx .934172$ is defined in Eq. [169], and $H = \tau/[3^{1/2}(1 + \tau)] = [(3 - 5^{1/2})/6]^{1/2} \approx 0.356822$ is defined in Eq. [170]. The sum indicated in parentheses is the sum of squares of the x, y, and z direction sums with normalized gradients.

$$g_x(n) = \cos\{(N\pi)^{1/2}\sin^{-1}[g_z(n)]\}(1 - g_z(n)^2)^{1/2} \quad [187]$$

$$g_y(n) = \sin\{(N\pi)^{1/2}\sin^{-1}[g_z(n)]\}(1 - g_z(n)^2)^{1/2} = [1 - g_z(n)^2 - g_x(n)^2]^{1/2} \quad [188]$$

where $n = 1, 2, \ldots, N$.

One other way to select well-spaced directions is to consider each gradient vector to represent two diametrically opposed point charges on a unit sphere, and to minimize the repulsive force between the charges (*47*). This does not yield nice analytic formulas, but should give optimal spacing for any number of directions.

Suboptimal Gradient Schemes

Sometimes it is useful to consider a suboptimal gradient encoding scheme to demonstrate the advantage of an optimal scheme. One example of such a suboptimal scheme, scheme 6x, is shown in Table 16 (*5*, *21*, *42*, *46*, *59*). Two suboptimal seven-direction schemes can be produced by combining scheme 6x with either (F, F, F) (*38*, *46*, *59*) or (F, F, $-F$) (see Table 16).

Evaluation of Gradient Encoding Schemes

There is no universal agreement on how to evaluate different gradient encoding schemes. This section briefly discusses several methods that have been suggested.

It has been suggested that minimizing the condition number would optimize the gradient encoding scheme (*57*). However, except for icosahedral gradient schemes, the condition number changes when the entire encoding scheme is rotated, so it is not rotationally invariant (*51*). Therefore, a lower condition number does not necessarily mean a better encoding scheme. If a condition number remains near the value for icosahedral schemes ($10^{1/2}/2 \sim 1.581$) when the entire gradient scheme is rotated, then the scheme is nearly icosahedral, and therefore nearly optimal. This article does not explore the definition of condition number and its relationship to encoding schemes. The condition number can be calculated with common programs such as IDL® using published definitions (*42*, *46*).

To avoid the problems of the condition number, Papadakis et al. suggested calculating the total variance of an isotropic tensor, the sum of the variances of the nine tensor elements (*59*). Hasan et al. modified this by normalizing each scheme to the variance with six icosahedral directions, and called the normalized factor the relative encoding advantage factor, REAF (*42*).

It would be convenient to have a gradient scheme evaluation that can be calculated with something as simple as common spreadsheet software. One simple measure of uniform spatial distribution is to calculate the dot product (see Eq. [I-3]) between each pair of normalized gradient vectors (gradient dot product, or *GDP*), which is inversely related to the angle θ between each pair of gradient vectors (*46*).

$$\mathbf{g}_n^{\mathrm{T}}\mathbf{g}_m = GDP = \cos\theta \quad [189]$$

$$GDP_{\max} = \cos\theta_{\min} \quad [190]$$

Table 14 The *x*, *y*, and *z* Components of Vectors Corresponding to 15 Gradient Directions Derived from the 30 Icosahedral or Dodecahedral Edges

	Scheme 15e (Sum = 0.167)														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
g_x	-1	0	0	C	C	C	C	$-.5$	.5	$-.5$	.5	$-D$	$-D$	D	D
g_y	0	1	0	D	D	$-D$	$-D$	$-C$	$-C$	$-C$	$-C$	$-.5$	.5	$-.5$	.5
g_z	0	0	1	.5	$-.5$	.5	$-.5$	$-D$	$-D$	D	D	$-C$	$-C$	$-C$	$-C$

Note: $C = (\tau - 1)/2 \approx 0.309017$ is defined in Eq. [165], and $D = \tau/2 \approx 0.809017$ is defined in Eq. [166]. The sum indicated in parentheses is the sum of squares of the x, y, and z direction sums with normalized gradients.

Table 15 Summary of Individual Polyhedral Encoding Schemes

Scheme	Polyhedron	Minimum Sum	*sRA* Directly (Eq. [177])?
3x	Cube faces; octahedral vertices	3	No
4t	Tetrahedral faces and vertices; cube vertices; octahedral faces	0	No
6p	Cube edges; octahedral edges	2	No
6v	Icosahedral vertices; dodecahedral faces	1.53	Yes
10f	Icosahedral faces; dodecahedral vertices	.584	Yes, if no noise
15e	Icosahedral edges; dodecahedral edges	.167	Yes, if no noise

Note: With each of these schemes, D_{av} equals the average of the individual *ADC*s (Eq. [175]).

Increased gradient spacing means that the largest *GDP*, GDP_{max}, becomes smaller, or the smallest angle θ, θ_{min}, becomes greater. Thus, improving spatial uniformity of gradient directions means maximizing θ_{min} or minimizing GDP_{max} among all the gradient directions. This provides one convenient way to compare two gradient schemes with equal numbers of gradients. Calculation of the dot product between the gradients of any scheme is easily performed on a spreadsheet.

Clearly, as the number of gradient directions, *M*, increases, some of the gradient directions must become closer together. Therefore, θ_{min} generally decreases as *M* increases, and GDP_{max} increases. Thus, although an increased number of encoding directions clearly provides more uniform spatial coverage, the value of θ_{min} does not reflect this improvement. Another criterion for evaluating gradient schemes is to calculate the furthest angular separation between any individual vector direction and the nearest gradient direction. There does not appear to be a systematic way to do this in the general case. One practical approach is to find a gradient direction whose θ_{min} is as large as possible, or whose GDP_{max} is as small as possible. One could then select a vector position by averaging the gradient direction and its closest neighbor, calculating the dot product of this normalized vector with each gradient direction (vector dot product, or *VDP*), then seeking to minimize VDP_{max} by adjusting the vector position slightly. This can be performed with the Solver in Excel. When the minimum value of VDP_{max} is found, this indicates the furthest possible angular distance between an arbitrary vector direction and the nearest gradient direction. This approach is similar to the proposed guaranteed anisotropic sensitivity, or *GAS*, which measures the range of VDP^2 for the worst possible case vector (*61*).

$$GAS = \mathrm{Min}(VDP^2_{max} - VDP^2_{min}) \qquad [191]$$

The preceding considerations considered only uniformity of spacing. In some cases, a small amount of spatial uniformity may be sacrificed for increased gradient strength, which can achieve the same *b* factor with a shorter *TE*. For example, scheme 6v is slightly more uniform than 6p, but 6p uses two gradients in the ratio 1:1 compared to a ratio 1:0.618 in 6v. Thus, for a given gradient duration δ and separation Δ, the *b* factor increases by a factor of $(1^2 + 1^2)/(1^2 + 0.618^2) \approx 2/1.38$. This advantage has been called *BMerit*, which was defined as "*b* divided by the *b* available using only one gradient coil" (*61*). Thus, *BMerit* is 2 for scheme 6p, and 1.38 for scheme 6v (Table 17).

Selecting a Gradient Encoding Scheme

There are several things to keep in mind when selecting a gradient sampling scheme. First, at least 20–30 images should be acquired with $b > 0$, so if fewer than 20 directions are included, each acquisition should be repeated. Repeating a direction with negative gradients allows the effects of nondiffusion gradients (static background gradients and imaging gradients) to be completely eliminated. However, this processing option does not yet appear to be available

Table 16 The Normalized *x*, *y*, and *z* Gradient Vector Components for a Suboptimal Six-Direction Gradient Sampling Scheme

	Scheme 6x					
	1	2	3	4	5	6
g_x	1	0	0	*E*	0	*E*
g_y	0	1	0	*E*	*E*	0
g_z	0	0	1	0	*E*	*E*

Note: $E = 2^{1/2}/2 \approx 0.707107$ is defined in Eq. [167]. Scheme 7x includes scheme 6x plus (*F*, *F*, *F*), and scheme 7y includes scheme 6x plus (*F*, *F*, −*F*), where $F = 3^{1/2}/3 \approx .577350$ is defined in Eq. [168].

Table 17 Evaluation Parameters for Several Gradient-Encoding Schemes

Scheme	*BMerit*[a]	GDP_{max}	Condition Number[a]	*GAS*	Vector for *GAS*	Minimum VDP_{max}	Vector for *VDP*
Near-optimal schemes							
3a	2.25	0	—	0	1, 1, 1	.577	1, 1, 1
4t	3	.333	—	0	1, 0, 0	.577	1, 0, 0
6p	2	.5	2	.5	1, 0, 0	.707	1, 0, 0
6v	1.38	.447	1.581	.596	1, 1, 1	.795	1, 1, 1
7c	1	.577	1.528	.651	1, .732, 0	.807	1, .732, 0
9p	1	.816	1.414	.667	1, 1, 1	.816	1, 1, 1
10f	1	.745	1.581	.596	1, .618, 0	.795	1, .618, 0
12u	1.25	.8	1.387	.53	1, 1, 1	.775	1, 1, 1
13o	1	.807	1.627	.754	1, .225, .507	.874	1, .225, .507
15e	1	.809	1.581	.72	1, .618, 0	.851	1, .618, 0
16c	1	.795	1.581	.85	1, 0, .159	.923	1, 0, .159
21c	1	.851	1.581	.86	1, .453, .618	.934	1, 1, 1
25c	1	.934	1.581	.85	1, .618, 0	.851	1, .618, 0
31c	1	.934	1.581	.85	1, .453, .791	.947	1, 1.34, .545
Suboptimal schemes							
6x	1	.707	2.618	.460	1, −1, .414	.679	1, −1, .414
7x	1	.816	2.562	.460	1, −1, .414	.679	1, −1, .414
7y	1	.816	2.59	.460	1, −1, .414	.679	1, −1, .414

[a] The condition number and *BMerit* may change if the gradient scheme is rotated. For any number of directions, more optimal schemes are characterized by high *BMerit*, *GAS*, and minimum VDP_{max}, and by a condition number near 1.581 even when the gradient scheme is rotated. For a given number of directions, more optimal schemes are characterized by lower GDP_{max}. The indicated vectors for *GAS* and minimum VDP_{max} are the worst case found so far, but it is possible in some cases that an even worse case could be found. These vectors are not normalized.

from commercial MRI vendors, so offline processing may be necessary. Second, the gradient directions should be well spaced out. Icosahedral schemes with 6, 10, 15, 16, 21, 25, or 31 directions provide well-spaced directions. Other reasonable schemes are available for 7, 9, 12, and 13 directions. Algorithms are available for selecting larger numbers of well-spaced directions. Third, the use of 21 or more directions will provide more uniform variance in the calculated DTI parameters over all possible tensor orientations and will minimize the effect of signals that approach the noise level with very high anisotropy levels. Fourth, if only six directions are used, scheme 6p (gradient pairs) allows the shortest *TE*, whereas scheme 6v (icosahedral vertices) retains some of the *TE* advantage with slightly more uniform spacing.

Important Points in "Selecting Gradient Directions"

It is generally best to have the gradient directions spread out uniformly in space. One convenient way to do this is to have the gradients point toward the vertices, edges, or faces of an icosahedron or other Platonic solid, or some combination of these. When the number of gradient directions does not allow an icosahedral sampling scheme, some good nonicosahedral analytic schemes are available. With certain sampling schemes, D_{av} and sometimes *sRA* can be calculated directly from the individual *ADC* measurements without explicitly calculating the tensor (Eqs. [175–178]). Several methods are available for selecting large numbers of directions that are spaced approximately uniformly. There is no universally accepted way to compare the effectiveness of different sampling schemes. Several evaluation criteria are compared in Table 17.

ACKNOWLEDGMENTS

This work was supported, in part, by an award from the Bachmann-Strauss Dystonia and Parkinson Foundation and by NIH grants R01-NS047668, R03-MH64554, R01-MH64823, R01-NS043448, and K01-MH65580. The author is grateful to W. Gordon Monahan for many useful discussions and suggestions during the preparation of this manuscript, to Jianhui Zhong for helpful comments on the manuscripts, and to the reviewers for several helpful suggestions.

REFERENCES

1. Bahn MM. 1999. Comparison of scalar measures used in magnetic resonance diffusion tensor imaging. J Magn Reson 139:1–7.
2. Alexander AL, Hasan K, Kindlmann G, Parker DL, Tsuruda JS. 2000. A geometric analysis of diffusion tensor measurements of the human brain. Magn Reson Med 44:283–291.
3. Basser PJ, Pierpaoli C. 1996. Microstructural and physiological features of tissues elucidated by quantitative-diffusion-tensor MRI. J Magn Reson B 111:209–219.
4. Bastin ME, Armitage PA, Marshall I. 1998. A theoretical study of the effect of experimental noise on the measurement of anisotropy in diffusion imaging. Magn Reson Imaging 16:773–785.
5. Papadakis NG, Xing D, Houston GC, Smith JM, Smith MI, James MF, et al. 1999. A study of rotationally invariant and symmetric indices of diffusion anisotropy. Magn Reson Imaging 17:881–892.
6. Hasan KM, Narayana PA. 2003. Computation of the fractional anisotropy and mean diffusivity maps without tensor decoding and diagonalization: Theoretical analysis and validation. Magn Reson Med 50:589–598.
7. Hasan KM, Alexander AL, Narayana PA. 2004. Does fractional anisotropy have better noise immunity characteristics than relative anisotropy in diffusion tensor MRI? An analytical approach. Magn Reson Med 51: 413–417.
8. Conturo TE, McKinstry RC, Akbudak E, Robinson BH. 1996. Encoding of anisotropic diffusion with tetrahedral gradients: A general mathematical diffusion formalism and experimental results. Magn Reson Med 35:399–412.
9. Armitage PA, Bastin ME. 2000. Selecting an appropriate anisotropy index for displaying diffusion tensor imaging data with improved contrast and sensitivity. Magn Reson Med 44:117–121.
10. Sorensen AG, Wu O, Copen WA, Davis TL, Gonzalez RG, Koroshetz WJ, et al. 1999. Human acute cerebral ischemia: Detection of changes in water diffusion anisotropy by using MR imaging. Radiology 212:785–792.
11. Basser PJ. 1995. Inferring microstructural features and the physiological state of tissues from diffusion-weighted images. NMR Biomed 8:333–344.
12. Skare S, Li T, Nordell B, Ingvar M. 2000. Noise considerations in the determination of diffusion tensor anisotropy. Magn Reson Imaging 18:659–669.
13. Armitage PA, Bastin ME. 2001. Utilizing the diffusion-to-noise ratio to optimize magnetic resonance diffusion tensor acquisition strategies for improving measurements of diffusion anisotropy. Magn Reson Med 45: 1056–1065.
14. Le Bihan D, Mangin JF, Poupon C, Clark CA, Pappata S, Molko N, Chabriat H. 2001. Diffusion tensor imaging: concepts and applications. J Magn Reson Imaging 13:534–546.
15. Chepuri NB, Yen YF, Burdette JH, Li H, Moody DM, Maldjian JA. 2002. Diffusion anisotropy in the corpus callosum. AJNR Am J Neuroradiol 23:803–808.
16. Dong Q, Welsh RC, Chenevert TL, Carlos RC, Maly-Sundgren P, Gomez-Hassan DM, Mukherji SK. 2004. Clinical applications of diffusion tensor imaging. J Magn Reson Imaging 19:6–18.
17. Horsfield MA, Jones DK. 2002. Applications of diffusion-weighted and diffusion tensor MRI to white matter diseases—a review. NMR Biomed 15:570–577.
18. Lim KO, Helpern JA. 2002. Neuropsychiatric applications of DTI—a review. NMR Biomed 15:587–593.
19. Masutani Y, Aoki S, Abe O, Hayashi N, Otomo K. 2003. MR diffusion tensor imaging: recent advance and new techniques for diffusion tensor visualization. Eur J Radiol 46:53–66.
20. Moseley M. 2002. Diffusion tensor imaging and aging—a review. NMR Biomed 15:553–560.
21. Ulug AM, van Zijl PCM. 1999. Orientation-independent diffusion imaging without tensor diagonalization: Anisotropy definitions based on physical attributes of the diffusion ellipsoid. J Magn Reson Imaging 9:804–813.
22. Kingsley PB, Monahan WG. 2005. Contrast-to-noise ratios of diffusion anisotropy indices. Magn Reson Med 53:911–918. [Erratum: Magn Reson Med 2005;54:251]
23. Green HAL, Peña A, Price CJ, Warburton EA, Pickard JD, Carpenter TA, Gillard JH. 2002. Increased anisotropy in acute stroke: A possible explanation. Stroke 33:1517–1521.
24. Sun SW, Song SK, Hong CY, Chu WC, Chang C. 2001. Improving relative anisotropy measurement using directional correlation of diffusion tensors. Magn Reson Med 46:1088–1092.
25. Pierpaoli C, Basser PJ. 1996. Toward a quantitative assessment of diffusion anisotropy. Magn Reson Med 36:893–906. [Published erratum: Magn Reson Med 1997;37:972.]
26. Poonawalla AH, Zhou XJ. 2004. Analytical error propagation in diffusion anisotropy calculations. J Magn Reson Imaging 19:489–498.
27. Westin CF, Peled S, Gudbjartsson H, Kikinis R, Jolesz FA. 1997. Geometrical diffusion measures for MRI from tensor basis analysis. ISMRM 5th Scientific Meeting, p 1742 (abstract).
28. Hasan KM, Basser PJ, Parker DL, Alexander AL. 2001. Analytical computation of the eigenvalues and eigenvectors in DT-MRI. J Magn Reson 152:41–47.
29. Neil JJ. 1997. Measurement of water motion (apparent diffusion) in biological systems. Concepts Magn Reson 9:385–401.
30. Price WS. 1997. Pulsed-field gradient nuclear magnetic resonance as a tool for studying translational diffusion: Part 1. Basic theory. Concepts Magn Reson 9:299–336.
31. Basser PJ. 2002. Diffusion and diffusion tensor MR imaging. In: Attard JJ, editor. Magnetic resonance imaging of the brain and spine. Philadelphia: Lippincott Williams & Wilkins. p 197–214.

32. Stejskal EO, Tanner JE. 1965. Spin diffusion measurements: Spin echoes in the presence of a time-dependent field gradient. J Chem Phys 42:288–292.
33. Stejskal EO. 1965. Use of spin echoes in a pulsed magnetic-field gradient to study anisotropic, restricted diffusion and flow. J Chem Phys 43:3597–3603.
34. Mulkern RV, Vajapeyam S, Robertson RL, Caruso PA, Rivkin MJ, Maier SE. 2001. Biexponential apparent diffusion coefficient parametrization in adult vs newborn brain. Magn Reson Imaging 19:659–668.
35. Mulkern RV, Gudbjartsson H, Westin CF, Zengingonul HP, Gartner W, Guttmann CRG, et al. 1999. Multi-component apparent diffusion coefficients in human brain. NMR Biomed 12:51–62.
36. Mulkern RV, Zengingonul HP, Robertson RL, Bogner P, Zou KH, Gudbjartsson H, et al. 2000. Multi-component apparent diffusion coefficients in human brain: relationship to spin-lattice relaxation. Magn Reson Med 44:292–300.
37. Clark CA, Le Bihan D. 2000. Water diffusion compartmentation and anisotropy at high b values in the human brain. Magn Reson Med 44:852–859.
38. Basser PJ, Mattiello J, LeBihan D. 1994. MR diffusion *tensor* spectroscopy and imaging. Biophys J 66:259–267.
39. Basser PJ, Mattiello J, LeBihan D. 1994. Estimation of the effective self-diffusion *tensor* from the NMR spin echo. J Magn Reson B 103:247–254.
40. Mattiello J, Basser PJ, Le Bihan D. 1997. The b matrix in diffusion tensor echo-planar imaging. Magn Reson Med 37:292–300.
41. Bammer R. 2003. Basic principles of diffusion-weighted imaging. Eur J Radiol 45:169–184.
42. Hasan KM, Parker DL, Alexander AL. 2001. Comparison of gradient encoding schemes for diffusion-tensor MRI. J Magn Reson Imaging 13:769–780.
43. Hsu EW, Mori S. 1995. Analytical expressions for the NMR apparent diffusion coefficients in an anisotropic system and a simplified method for determining fiber orientation. Magn Reson Med 34:194–200.
44. Neeman M, Freyer JP, Sillerud LO. 1991. A simple method for obtaining cross-term-free images for diffusion anisotropy studies in NMR microimaging. Magn Reson Med 21:138–143.
45. Güllmar D, Haueisen J, Reichenbach JR. 2005. Analysis of *b*-value calculations in diffusion weighted and diffusion tensor imaging. Concepts Magn Reson 25A: 53–66.
46. Hasan KM, Parker DL, Alexander AL. 2002. Magnetic resonance water self-diffusion tensor encoding optimization methods for full brain acquisition. Image Anal Stereol 21:87–96.
47. Jones DK, Horsfield MA, Simmons A. 1999. Optimal strategies for measuring diffusion in anisotropic systems by magnetic resonance imaging. Magn Reson Med 42:515–525.
48. Press WH, Teukolsky SA, Vetterling WT, Flannery BP. 1997. Numerical recipes in C. Cambridge: Cambridge University Press.
49. Alexander DC, Barker GJ. 2005. Optimal imaging parameters for fiber-orientation estimation in diffusion MRI. Neuroimage 27:357–367.
50. Jones DK. 2004. The effect of gradient sampling schemes on measures derived from diffusion tensor MRI: A Monte Carlo study. Magn Reson Med 51:807–815.
51. Batchelor PG, Atkinson D, Hill DLG, Calamante F, Connelly A. 2003. Anisotropic noise propagation in diffusion tensor MRI sampling schemes. Magn Reson Med 49:1143–1151.
52. Papadakis NG, Murrills CD, Hall LD, Huang CLH, Carpenter TA. 2000. Minimal gradient encoding for robust estimation of diffusion anisotropy. Magn Reson Imaging 18:671–679.
53. Akkerman EM. 2003. Efficient measurement and calculation of MR diffusion anisotropy images using the Platonic variance method. Magn Reson Med 49:599–604.
54. Basser PJ, Pierpaoli C. 1998. A simplified method to measure the diffusion tensor from seven MR images. Magn Reson Med 39:928–934.
55. Özcan A. 2005. (Mathematical) Necessary conditions for the selection of gradient vectors in DTI. J Magn Reson 172:238–241.
56. Conturo TE, McKinstry RC, Aronovitz JA, Neil JJ. 1995. Diffusion MRI: Precision, accuracy and flow effects. NMR Biomed 8:307–332.
57. Skare S, Hedehus M, Moseley ME, Li TQ. 2000. Condition number as a measure of noise performance of diffusion tensor data acquisition schemes with MRI. J Magn Reson 147:340–352.
58. Naganawa S, Koshikawa T, Kawai H, Fukatsu H, Ishigaki T, Maruyama K, Takizawa O. 2004. Optimization of diffusion-tensor MR imaging data acquisition parameters for brain fiber tracking using parallel imaging at 3 T. Eur Radiol 14:234–238.
59. Papadakis NG, Xing D, Huang CLH, Hall LD, Carpenter TA. 1999. A comparative study of acquisition schemes for diffusion tensor imaging using MRI. J Magn Reson 137:67–82.
60. Wong STS, Roos MS. 1994. A strategy for sampling on a sphere applied to 3D selective RF pulse design. Magn Reson Med 32:778–784.
61. Muthupallai R, Holder CA, Song AW, Dixon WT. 1999. Navigator aided, multishot EPI diffusion images of brain with complete orientation and anisotropy information. ISMRM 7th Scientific Meeting, p 1825 (abstract).

BIOGRAPHY

Peter B. Kingsley, MRI Research Physicist at North Shore University Hospital in Manhasset, NY, became fascinated with NMR during his undergraduate studies at Dartmouth College (B.A., 1974). At Cornell University (Ph.D., 1980) he extended the concept of "deuterated solvent" by preparing perdeuterated phospholipids for ^{1}H NMR studies of lipid-soluble compounds. After some ^{31}P NMR studies of perfused heart metabolism at the University of Minnesota (1985–1991), he helped establish a human MR spectroscopy program at St. Jude Children's Research Hospital in Memphis, Tennessee, before moving to Long Island in 1996. His current research and clinical interests include diffusion tensor imaging, MR spectroscopy, and T_1 and T_2 mapping.