

10 SRF operations

10.1 Introduction

This International Standard specifies operations on SRF coordinates and, in the case of 3D object-spaces, on SRF spatial directions, vectors and orientations. Underlying these operations are the similarity transformations relating two ORMs. Similarity transformations are treated first in [10.3](#). The general case of changing the representation of a position as a coordinate in one SRF to its representation as a coordinate in another SRF is specified in [10.4](#), followed by important special cases. The specification of a spatial direction, vector or orientation in the context of an SRF is defined, and operations for changing these representations from one SRF to their corresponding representations in another SRF are specified in [10.5](#).

Euclidean distance in 2D and 3D object-space is specified in [10.6](#). Geodesic distance and azimuth on the surface of an oblate ellipsoid (or sphere) are specified in [10.7](#).

10.2 Notation and terminology

An important category of spatial operations is changing the representation of spatial information in one SRF to the representation in a second SRF. For these change of SRF operations, the adjective “source” shall be used to refer to the first SRF, and the adjective “target” shall be used to refer to the second SRF.

The notation in [Table 10.1](#) is used throughout this clause.

Table 10.1 — Notation

Notation	Description
ORM_S	Source ORM
ORM_T	Target ORM
ORM_R	Reference ORM for a given spatial object
H_{RT}	Reference transformation from the reference ORM_R to ORM_T
H_{SR}	Reference transformation from ORM_S to the reference ORM_R
H_{TR}	Reference transformation from ORM_T to the reference ORM_R
H_{ST}	Similarity transformation from the embedding of ORM_S to ORM_T
M_{RT}	Rotation matrix from the reference ORM_R to ORM_T
M_{SR}	Rotation matrix from ORM_S to the reference ORM_R
M_{ST}	Rotation matrix from ORM_S to ORM_T
M_{TR}	Rotation matrix from ORM_T to the reference ORM_R
SRF_S	Source SRF based on ORM_S
SRF_T	Target SRF based on ORM_T
SRF_L	The local tangent frame SRF at a coordinate (See 10.5.2)
CS_S	CS of SRF_S
CS_T	CS of SRF_T
G_S	Generating function of CS_S
G_T^{-1}	Inverse generating function of CS_T
c_S	Coordinate of a spatial position in SRF_S

c_T	Coordinate of a spatial position in SRF _T
n_S	Direction vector in SRF _S (See 10.5.2)
n_T	Direction vector in SRF _T (See 10.5.2)
R_{ST}	Orientation of SRF _T with respect to SRF _S in the position vector rotation convention

10.3 Operations on ORMs

10.3.1 Introduction

The similarity transformation (see [7.3.2](#)) H_{ST} between a source/target pair ORM_S and ORM_T underlies the coordinate operations in [10.4](#). Given a set of n ORMs there are $n(n-1)$ such source and target ORM pairs. Instead of specifying the full set of similarity transformations, this International Standard reduces the requirement to specifying the reference transformation H_{SR} from each object-fixed source ORM_S to the reference ORM_R for a given object. This subclause specifies the methods of expressing a similarity transformation H_{ST} in terms of the reference transformations for the source and target ORMs. The cases of ORMs for a single object are treated in [10.3.2](#). The more general cases in which ORM_S and ORM_T are ORMs for different objects are treated in [10.3.3](#).

10.3.2 ORMs for a single object

If ORM_S is an object-fixed ORM, its reference transformation H_{SR} is a type of similarity transformation. Any 3D or 2D similarity transformation may be represented with the STT [ROTATE SCALE TRANSLATE](#) in the 3D case or STT [ROTATE SCALE TRANSLATE 2D](#) in the 2D case (see Tables [7.19](#) and [7.20](#)). Thus using the notation of the STT formulation, H_{SR} may be represented in the form given by [Equation \(10.1\)](#).

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_R = H_{SR} \begin{pmatrix} x \\ y \\ z \end{pmatrix}_S \equiv \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix}_{SR} + s_{SR} M_{SR} \begin{pmatrix} x \\ y \\ z \end{pmatrix}_S \quad (10.1)$$

NOTE The processes by which ORMs for the Earth are established are based on physical measurements. These measurements are subject to error, and therefore introduce various types of relative distortions between ORMs. [Equation \(10.1\)](#) is based on the assumption that positions in object-space are error free, and the equation includes no compensation for these distortions.

The reference transformation H_{TR} from ORM_T to the reference ORM_R is also a similarity transformation.

An important operation is the similarity transformation H_{ST} from ORM_S to ORM_T, when neither the source nor the target is necessarily the reference ORM. The H_{ST} transformation may be expressed as the composition of H_{SR} with H_{TR}^{-1} (or H_{RT} , which is equivalent to the inverse of H_{TR}) as in [Equation \(10.2\)](#) (see [Figure 10.1](#)):

$$H_{ST} = H_{RT} \circ H_{SR} \quad (10.2)$$

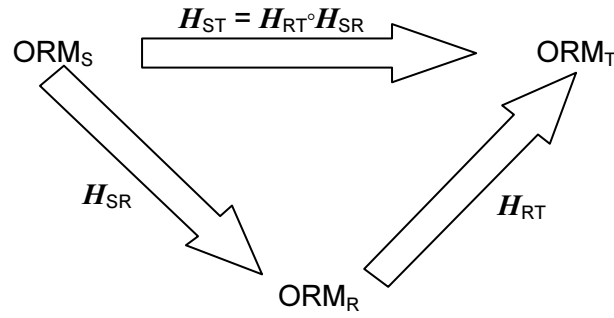


Figure 10.1 — Composed transformations

H_{RT} is also a similarity transformation:

$$\begin{aligned} H_{RT} \begin{pmatrix} x \\ y \\ z \end{pmatrix}_R &= \frac{1}{s_{TR}} M_{TR}^{-1} \begin{pmatrix} x \\ y \\ z \end{pmatrix}_R - \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix}_{ST} \\ &= \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix}_{RT} + \frac{1}{s_{TR}} M_{TR}^{-1} \begin{pmatrix} x \\ y \\ z \end{pmatrix}_R \end{aligned}$$

Because the matrix M_{TR} is a rotation matrix, its transpose M_{TR}^T is also its inverse M_{TR}^{-1} . The inverse of M_{TR} is also the matrix M_{RT} corresponding to the reverse rotations of ORM_T with respect to ORM_R . In particular:

$$M_{RT} = M_{TR}^{-1} = M_{TR}^T$$

and

$$H_{RT} \begin{pmatrix} x \\ y \\ z \end{pmatrix}_R = \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix}_{RT} + \frac{1}{s_{TR}} M_{RT} \begin{pmatrix} x \\ y \\ z \end{pmatrix}_R.$$

The composite operation $H_{ST} = H_{RT} \circ H_{SR}$ reduces to:

$$H_{ST} \begin{pmatrix} x \\ y \\ z \end{pmatrix}_S = H_{RT} \circ H_{SR} \begin{pmatrix} x \\ y \\ z \end{pmatrix}_S = \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix}_{ST} + \frac{s_{SR}}{s_{TR}} M_{ST} \begin{pmatrix} x \\ y \\ z \end{pmatrix}_S \quad (10.3)$$

where:

$$\begin{aligned} M_{ST} &= M_{RT} \circ M_{SR}, \text{ and} \\ \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix}_{ST} &= \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix}_{RT} + \frac{1}{s_{TR}} M_{RT} \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix}_{SR}. \end{aligned}$$

If the rotations M_{SR} and M_{TR} are equal, then M_{ST} is the identity matrix, and if $s_{SR} = s_{TR}$, H_{ST} simplifies to a translation of the origin:

$$H_{ST} \begin{pmatrix} x \\ y \\ z \end{pmatrix}_S = \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix}_{ST} + \begin{pmatrix} x \\ y \\ z \end{pmatrix}_S.$$

[Equation \(10.2\)](#) and [Figure 10.1](#) also apply to the 2D case.

If the source ORM_S is a time-dependent ORM for a spatial object, ORM_S(*t*) shall denote the ORM_S at time *t*, and $H_{SR}(t)$ shall denote the similarity transformation from ORM_S(*t*) to the object-fixed reference ORM_R. If the similarity transformation $H_{SR}(t)$ can be determined, it is a time-dependent affine transformation. For a fixed value of time *t*₀, [Equation \(10.2\)](#) and [Figure 10.1](#) generalize to $H_{ST}(t_0) = H_{RT} \circ H_{SR}(t_0)$. The generalization to a time-dependent target ORM_T(*t*) is $H_{ST}(t_0) = H_{RT}(t_0) \circ H_{SR}$. The generalization when both ORMs are time-dependent at time *t*₀ is $H_{ST}(t_0) = H_{RT}(t_0) \circ H_{SR}(t_0)$.

EXAMPLE ORM_S(*t*) is the ORM [EARTH INERTIAL J2000r0](#) at time *t*. ORM_R is the Earth reference ORM [WGS 1984](#). Because ORM_S(*t*) and ORM_R share the same embedding origin, the $H_{SR}(t)$ transformation is a (rotation) matrix multiplication operation (without translation). The matrix coefficients for selected values of *t* account for polar motion, Earth rotation, nutation, and precession. Predicted values for these coefficients are computed and updated weekly by the International Earth Rotation Service (IERS) [\[IERS\]](#). See [7.5](#) for other examples of dynamic ORM reference transformations.

10.3.3 Relating ORMs for different objects

If a spatial object **S** exists in the space of another spatial object **T**, and if ORM_R is the reference ORM for object **T**, and if the two objects are fixed with respect to each other, then H_{SR} shall denote a similarity transformation from ORM_S to ORM_R. H_{SR} is an affine transformation. If ORM_T is an object-fixed ORM for the object **T**, then H_{ST} is given by [Equation \(10.2\)](#). The time dependent generalizations of [Equation \(10.2\)](#), defined in [10.3.2](#), are also applicable to this case.

EXAMPLE ORM_S is an ORM for the space shuttle (as a spatial object). ORM_R is the Earth reference ORM [WGS 1984](#). When in orbit at time *t*, $H_{SR}(t)$ transforms positions with respect to ORM_S to positions with respect to ORM [WGS 1984](#).

If a spatial object **S** does not exist in the space of another spatial object **T**, a similarity transformation between their ORMs is not intrinsically determined. However, if an invertible affine transformation (H_{SR}) between ORM_S and the reference ORM for object **T** is provided, then, given an object-fixed ORM for object **T**, ORM_T, [Equation \(10.2\)](#) may be used to define an invertible affine transformation H_{ST} , from ORM_S to ORM_T. An important instance of this case occurs when **S** is an abstract object and **T** is a physical object (see [10.4.6](#)).

10.4 Operations to change spatial coordinates between SRFs

10.4.1 Introduction

Given a coordinate c_S in a source SRF, SRF_S, the change of SRF operation²⁵ computes the corresponding coordinate c_T in a given target SRF, SRF_T. The general case of this operation is presented in formulations in [10.4.2](#) for time-independent (static) and time-dependent (dynamic) ORM relationships. The specific SRF coordinate-systems CS_S and CS_T impose restrictions on the applicability of the formulation because of CS domain/range constraints or because of optional valid region constraints (see below).

The formulations depend on the existence of a (static or dynamic) embedding transformation H_{ST} from ORM_S to ORM_T. If ORM_S and ORM_T have the same object space, H_{ST} is formulated in [10.3.2](#) in terms of ORM specification elements. In the case of different object spaces, H_{ST} must be explicitly provided (see [10.3.2](#)).

Special cases allow for simplifications that result in computational short cuts to the general change of SRF formulation. The case of matched normal embeddings (which includes the case ORM_S = ORM_T) is treated in [10.4.3](#). Further specializations arise from combinations of specific coordinate-systems. Subclause [10.4.4](#) treats combinations of celestiodetic with a map projection.

²⁵ [ISO 19111](#) defines this case as a coordinate operation.

The case for which $CS_S = CS_T$ and ORM_S and ORM_T differ²⁶ does not generally produce a computational simplification. However, when both the source and target SRFs are based on the CS [LOCOCENTRIC EUCLIDEAN 3D](#), a simplification is produced and is presented in [10.4.5](#). This case is important for operations on directions, vectors, and orientations (see [10.5](#)).

An important special case of unrelated object spaces occurs when the source object space is an abstract 3D object space. This special case is treated in [10.4.6](#).

10.4.2 General case

SRF_S and SRF_T are two object-fixed SRFs for a spatial object and p is a point in object-space that is in the coordinate system domains of both SRFs. c_S denotes the coordinate of p in SRF_S , and c_T denotes the coordinate of p in SRF_T . The determination of c_T from c_S is an operation involving the SRF pair (SRF_S, SRF_T) . The most general form of the operation is:

$$c_T = G_T^{-1} \circ H_{ST} \circ G_S(c_S) \quad (10.4)$$

See [Figure 10.2](#). CS generating and inverse generating functions are specified in [Clause 5](#).

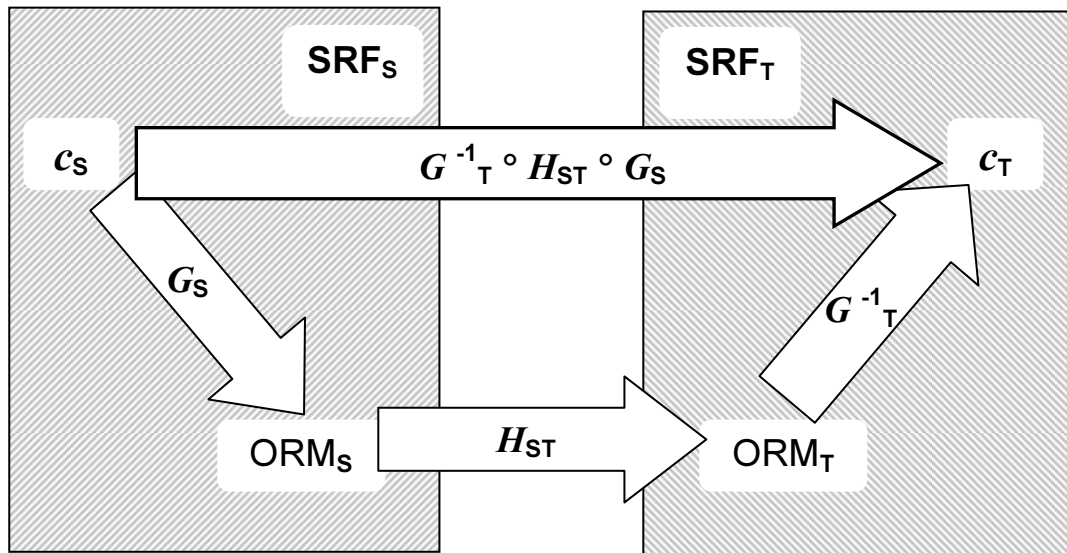


Figure 10.2 — Change of SRF operation – applied to coordinates

When H_{ST} is approximated with the Bursa-Wolf equation (see STT [PV 7 PARAMETER](#) Note 2), [Equation \(10.4\)](#) is known as the *Helmert transformation*.

²⁶ [ISO 19111](#) defines this case as a coordinate transformation.

If SRF_S and SRF_T are two celestiodetic SRFs with different ORMs for the same spatial object, [Equation \(10.4\)](#) transforms the coordinate $c_S = (\lambda_S, \varphi_S, h_S)$ with respect to one oblate ellipsoid to $c_T = (\lambda_T, \varphi_T, h_T)$ with respect to the other oblate ellipsoid. A transformation between two [celestiodetic](#) SRFs for the spatial object Earth is known as a *horizontal datum shift*.

NOTE A number of numerical approximations developed to implement horizontal datum shift have been published. Under the assumption of zero rotations and no scale differences, a widely used approximation²⁷ to directly transform $c_S = (\lambda_S, \varphi_S, h_S)$ to $c_T = (\lambda_T, \varphi_T, h_T)$ is the *standard Molodensky transformation* formula (see [\[83502T\]](#)).

In the time-dependent case, [Equation \(10.4\)](#) may be generalized to:

$$c_T(t) = G_T^{-1} \circ H_{ST}(t) \circ G_S(c_S)$$

[Equation \(10.4\)](#) is only defined for a value of c_S in the CS_S domain if its corresponding position belongs to the CS_T range. If D_S^{-1} is the domain of the inverse generating function G_S^{-1} and D_T^{-1} is the domain of the inverse generating function G_T^{-1} , [Equation \(10.4\)](#) is only defined for c_S in the set:

$$G_S^{-1}(D_S^{-1} \cap H_{ST}^{-1}(D_T^{-1})) \equiv \{c_S \text{ in } D_S \mid H_{ST}(G_S(c_S)) \text{ in } D_T^{-1}\} \quad (10.5)$$

EXAMPLE SRF_S is SRF [GEOCENTRIC_WGS_1984](#) and SRF_T is an instance of SRFT [MERCATOR](#), with ORM [WGS_1984](#). [Equation \(10.4\)](#) is not defined for any c_S that is on the z -axis of SRF_S , because the z -axis is not contained in the set in [Equation \(10.5\)](#).

SRF_T may optionally specify a valid-region V_T , and may optionally specify an extended-valid region E_T (see [8.3.2.4](#)). If D_T is the domain of the generating function G_T , then $V_T \subseteq E_T \subseteq D_T$. If [Equation \(10.4\)](#) is defined for c_S , c_T may be valid (c_T is in V_T), or extended valid (c_T is in $E_T \setminus V_T$) or defined (c_T is in $D_T \setminus E_T$) or invalid. The set of c_S coordinates for which c_T is valid is:

$$G_S^{-1}(D_S^{-1} \cap H_{ST}^{-1}(G_T(V_T))) \equiv \{c_S \text{ in } D_S \mid H_{ST}(G_S(c_S)) \text{ in } G_T(V_T)\}$$

where:

$$G_T(V_T) \equiv \{p \text{ in } D_T^{-1} \mid G_T^{-1}(p) \text{ in } V_T\}.$$

In applications that functionally conform to an SRM profile, the domain of an SRF operation is restricted to the accuracy domain of the SRF as specified by that profile (see [Clause 12](#)).

10.4.3 The matched normal embeddings case

If both source and target ORMs are the same, or, more generally, if the reference transformations of ORM_S and ORM_T are equivalent (*i.e.*, matched normal embeddings), H_{ST} is the identity transformation. Consequently, [Equation \(10.4\)](#) simplifies to:

$$c_T = G_T^{-1} \circ G_S(c_S). \quad (10.6)$$

EXAMPLE 1 If SRF_S is a [celestiodetic](#) SRF (see [8.4](#)) and SRF_T is the [celestiocentric](#) SRF for the same ORM ($\text{ORM}_S = \text{ORM}_T$), then since the CS of the [celestiocentric](#) SRF is [Euclidean_3D](#) for which the G_T^{-1} is the identity, [Equation \(10.6\)](#) reduces to the geodetic generating function: $c_T = G_S(c_S)$.

²⁷ Historically it was thought that these approximations would require less computation than direct conversion. The perceived computational advantage may have been overcome by technology advances. New efficient algorithms for converting celestiocentric coordinates to celestiodetic coordinates have been developed that result in appreciably faster transformations without the attendant loss of accuracy.

If SRF_T is a 3D SRF that has ellipsoidal height designated as the vertical coordinate-component of the SRF (see 8.4), and SRF_S is the induced zero height surface SRF, the *promotion operation* converts a surface coordinate c_S in SRF_S to a 3D coordinate in SRF_T by setting the 1st and 2nd coordinate-components of c_T to the 1st and 2nd coordinate-components of c_S and setting the 3rd coordinate-component, ellipsoidal height, to 0. Coordinate promotion is a special case of Equation (10.6). Applicable SRFs include those based on SRFT [CELESTIODETTIC](#), [PLANETODETTIC](#), and all map projection SRFTs.

EXAMPLE 2 If SRF_S is an induced surface [celestiodetic](#) SRF (see 8.4) and SRF_T is the 3D [celestiodetic](#) SRF for the same ORM ($\text{ORM}_S = \text{ORM}_T$), Equation (10.6) promotes $c_S = (\lambda, \varphi)$ from a coordinate of CS type surface to $c_T = (\lambda, \varphi, 0)$ a coordinate of CS type 3D.

If SRF_S is a 3D SRF that has ellipsoidal height designated as the vertical coordinate-component of the SRF (see 8.4), and SRF_T is the induced zero height surface SRF, the *truncation operation* converts a 3D coordinate c_S in SRF_S to a surface coordinate c_T , by setting the 1st and 2nd coordinate-components of c_T to the 1st and 2nd coordinate-components of c_S . The point in object-space corresponding to c_S and the point in object-space corresponding to c_T are not the same point unless $h = 0$. Truncation, therefore, does not generally preserve location.

EXAMPLE 3 If SRF_S is a [celestiodetic](#) 3D SRF, the (induced) surface SRF_T is the [surface celestiodetic](#) SRF for the same ORM. The truncation operation associates $c_T = (\lambda, \varphi)$ to $c_S = (\lambda, \varphi, h)$.

10.4.4 Matched normal embeddings and map projection SRFs

The CS generating function G_{MP} for a an augmented map projection SRF is implicitly defined (see 5.8.6) by the composition of the generating function for the [geodetic 3D](#) CS generating function G_{GD} with the inverse mapping equation $Q \equiv (Q_1, Q_2, h)$ as:

$$G_{MP} = G_{GD} \circ Q.$$

If SRF_S and SRF_T are map projection SRFs for the same object, and the corresponding reference transformations are equivalent, then Equation (10.6) becomes:

$$\begin{aligned} c_T &= (G_{GD, T} \circ Q_T)^{-1} \circ (G_{GD, S} \circ Q_S)(c_S) \\ &= P_T \circ G_{GD, T}^{-1} \circ G_{GD, S} \circ Q_S(c_S) \end{aligned} \quad (10.7)$$

where:

- Q_S : inverse mapping equations for SRF_S ,
- $G_{GD, S}$: generating function for the geodetic 3D CS for SRF_S ,
- Q_T : inverse mapping equations for SRF_T ,
- P_T : mapping equations for SRF_T , and
- $G_{GD, T}$: generating function for the geodetic 3D) CS for SRF_T .

Furthermore, if $\text{ORM}_S = \text{ORM}_T$, then $G_{GD, S} = G_{GD, T}$ and Equation (10.7) simplifies to:

$$c_T = P_T \circ Q_S(c_S). \quad (10.8)$$

If SRF_T is a [celestiodetic](#) SRF and $\text{ORM}_T = \text{ORM}_S$, Equation (10.6) simplifies to:

$$c_T = Q_S(c_S).$$

Similarly, if SRF_S is a [celestiodetic](#) SRF and $\text{ORM}_T = \text{ORM}_S$, Equation (10.6) simplifies to:

$$c_T = P_T(c_S).$$

10.4.5 Linear orthonormal 3D SRFs

The special case of source and target SRFs based on the CS [LOCOCENTRIC EUCLIDEAN 3D](#) is important for the treatment of directions, vectors, and orientations (see [10.5](#)). Every linear orthonormal CS may be viewed as an instance of a CS [LOCOCENTRIC EUCLIDEAN 3D](#). If SRF_S and SRF_T are two such SRFs (see [Table 8.11](#)), and F_{LE3D} is the CS [LOCOCENTRIC EUCLIDEAN 3D](#) (see [Table 5.9](#)) generating function, F_{LE3D} may be expressed in terms of the CS binding parameter vectors q, r, s in the form of the affine transformation:

$$\begin{aligned} F_{\text{LE3D}}(c) &= F_{\text{LE3D}} \begin{pmatrix} u \\ v \\ w \end{pmatrix} \\ &= q + ur + vs + wt \\ &= q + u \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix} + v \begin{pmatrix} s_1 \\ s_2 \\ s_3 \end{pmatrix} + w \begin{pmatrix} t_1 \\ t_2 \\ t_3 \end{pmatrix} \\ &= q + R c \end{aligned}$$

where:

$$\begin{aligned} t &= (r \times s), \text{ and} \\ R &= \begin{pmatrix} r_1 & s_1 & t_1 \\ r_2 & s_2 & t_2 \\ r_3 & s_3 & t_3 \end{pmatrix}. \end{aligned}$$

The inverse generating function is expressed as:

$$F_{\text{LE3D}}^{-1} \begin{pmatrix} u \\ v \\ w \end{pmatrix} = R^T \begin{pmatrix} u \\ v \\ w \end{pmatrix} - q$$

where: R^T is the transpose of R .

If vectors q_S, r_S, s_S, q_T, r_T , and s_T are the CS binding parameters (see [Table 8.11](#)) for SRF_S and SRF_T respectively, then substituting the expression in [Equation \(10.3\)](#) for H_{ST} , [Equation \(10.4\)](#) specializes to:

$$\begin{aligned} c_T &= F_{\text{LE3D}, T}^{-1} \circ H_{\text{ST}} \circ F_{\text{LE3D}, S}(c_S) \\ &= R_T^T (H_{\text{ST}}(q_S + R_S c_S) - q_T) \\ &= R_T^T \left(\begin{pmatrix} x \\ y \\ z \end{pmatrix}_{\text{ST}} + \begin{pmatrix} s_{\text{SR}} \\ s_{\text{TR}} \end{pmatrix} M_{\text{ST}} q_S \right) - q_T + \frac{s_{\text{SR}}}{s_{\text{TR}}} R_T^T \circ M_{\text{ST}} \circ R_S c_S. \end{aligned} \quad (10.9)$$

In the case that the corresponding reference transformations of ORM_S and ORM_T are equivalent, [Equation \(10.6\)](#) specializes to [Equation \(10.10\)](#):

$$\begin{aligned} c_T &= F_{\text{LE3D}, T}^{-1} \circ F_{\text{LE3D}, S}(c_S) \\ &= R_T^T (q_S - q_T) + R_T^T \circ R_S c_S. \end{aligned} \quad (10.10)$$

10.4.6 Instantiating abstract space linear SRFs

Engineering designs and abstract models are often intended for realization in the physical world.

EXAMPLE A building plan is designed in the source SRF_S, an abstract space [LOCAL SPACE RECTANGULAR 3D](#) SRF. A terrestrial site survey establishes the origin of the target SRF_T, a [LOCAL TANGENT SPACE EUCLIDEAN](#) SRF. Source coordinates are related to target coordinates by: $(x_T, y_T, z_T) = (1 + \Delta s)(x_S, y_S, z_S)$ where $(1 + \Delta s)$ is a scale factor.

More generally, models are scaled, rotated, or otherwise transformed by an invertible matrix 3×3 W before a source coordinate is associated to a target coordinate. In many application domains, this similarity transformation is in the form:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}_T = \begin{pmatrix} x_\Delta \\ y_\Delta \\ z_\Delta \end{pmatrix} + kW \begin{pmatrix} x \\ y \\ z \end{pmatrix}_S$$

where $k = (1 + \Delta s)$ is the scale factor, $(x_\Delta \ y_\Delta \ z_\Delta)$ is the translation displacement vector, and W is a rotation matrix. In the computer graphics application domain this transformation is often represented in matrix 4×4 form:

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}_T = \begin{pmatrix} a_{11} & a_{12} & a_{13} & x_\Delta \\ a_{21} & a_{22} & a_{23} & y_\Delta \\ a_{31} & a_{32} & a_{33} & z_\Delta \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}_S, \text{ where } \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} = kW.$$

This transformation between source and target coordinates may be viewed as an SRF coordinate operation from c_S in SRF_S, an abstract space [LOCAL SPACE RECTANGULAR 3D](#) SRF, to a coordinate c_T in SRF_T, a physical world [LOCOCENTRIC EUCLIDEAN 3D](#) SRF.

In the notation of [10.4.5](#):

$$\begin{aligned} G_S(c_S) &= R_S c_S, \text{ and} \\ G_T(c_T) &= q_T + R_T c_T. \end{aligned}$$

Define an invertible affine transformation H_{ST} as $H_{ST}(v) = q_T + R_T \circ W v$ (see [10.3.3](#)). Substitute this H_{ST} in [Equation \(10.4\)](#) and simplify:

$$\begin{aligned} c_T &= G_T^{-1} \circ H_{ST} \circ G_S(c_S) \\ &= R_T^T (H_{ST}(R_S c_S) - q_T) \\ &= R_T^T (q_T + R_T \circ W \circ R_S c_S - q_T) \\ &= W \circ R_S c_S \end{aligned} \tag{10.11}$$

This illustrates that the transformation $c_T = W \circ R_S c_S$ may be viewed as a change of SRF operation.

NOTE [Equation \(10.11\)](#) illustrates that digital graphic composite pattern modelling techniques such as SceneGraph trees that use scale and rotation matrices W together with translation operations at each tree node are special cases of [Equation \(10.4\)](#). See also [10.5.5 Example 2](#).

10.5 Operations on directions, vectors, and orientations

10.5.1 Introduction

Specification of 3D directions, vectors, or orientations associated with a 3D SRF requires an underlying 3D vector space. An SRF is either linear or curvilinear. In the linear cases, the structure of the coordinate-space provides such a 3D vector space. In particular, all lines through distinct points in a given direction \mathbf{n} are parallel in both coordinate- and object-space. This shows that a linear SRF supports the translation invariance of directions and vectors. A linear SRF will not preserve angular relationships between directions unless the associated abstract coordinate system (CS) is also orthonormal. In the orthonormal case, angles and distances are preserved.

In the case of a curvilinear 3D SRF, the structure of the coordinate-space does not provide an underlying 3D vector space. To support curvilinear 3D SRFs, a method of associating a 3D vector space with any given reference point in the SRF shall be used. This 3D vector space is termed the local tangent frame SRF at the reference point. This association of a local tangent frame with a reference point is applied uniformly to both curvilinear and linear SRFs.

The coordinate-space of an augmented map projection SRF (a map projection augmented with ellipsoidal height as a third dimension) appears to inherit the vector-space structure of \mathbf{R}^3 , however, the vector properties of the (easting, northing, height)-coordinates do not carry over to object-space. This is illustrated in part by the “up pointing” vector $\mathbf{n} = (0, 0, 1)$ that points in different spatial directions in object-space depending on the map coordinate location at which \mathbf{n} is placed.

In [Figure 10.3](#), distinct position points p and q on the ellipsoid surface are projected to augmented map coordinates $(s, t, 0)$ and $(u, v, 0)$. Starting at these map coordinates, the coordinates one unit away in the “up direction” are $(s, t, 1)$ and $(u, v, 1)$, respectively. In an augmented map projection, these coordinates correspond to the position-space points p' and q' . The direction from p to p' is not the same as the direction from q to q' . This shows that, in object-space, the “up direction” is relative to a reference point.

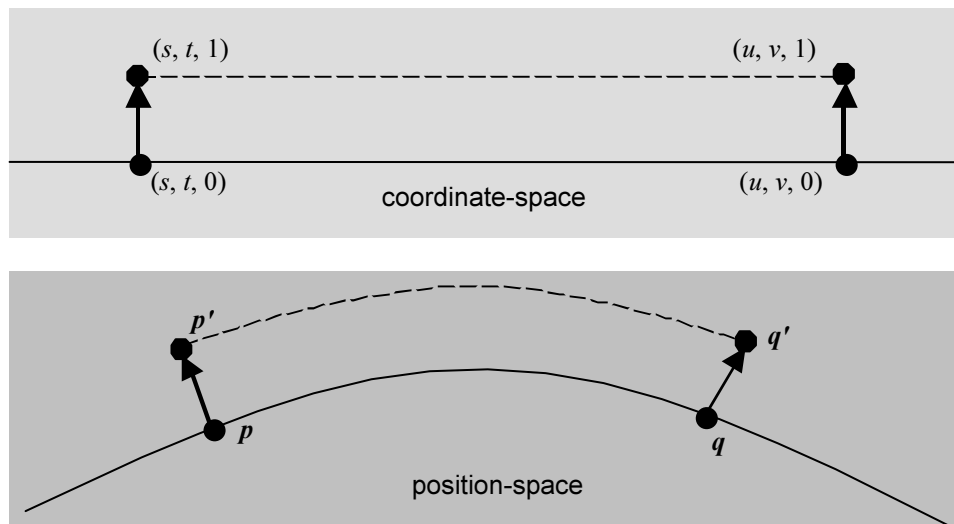


Figure 10.3 — Coordinate-space and position-space directions compared

A local tangent frame SRF, associated with a given reference point, shall be used to specify directions relative to that reference point. Such an SRF is defined by having its origin at the reference point and its axes given by the normalized vectors tangent to the coordinate curves passing through the reference point, as illustrated in [Figure 10.4](#). All linear and curvilinear CSs in this International Standard are orthogonal CSs, thus the local tangent frame is an orthonormal linear SRF.

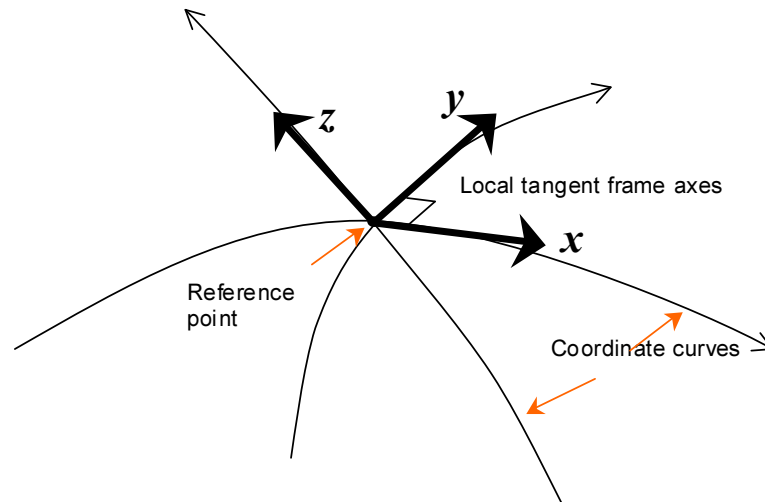


Figure 10.4 — Local tangent frame axes

[Figure 10.5](#) shows two local tangent frames at points p and q . The local "up" directions may be specified as a direction in either local tangent frame. Since directions are translation invariant in linear SRFs, conceptually the two local tangent frames may be translated to a common origin, as in [Figure 10.6](#).

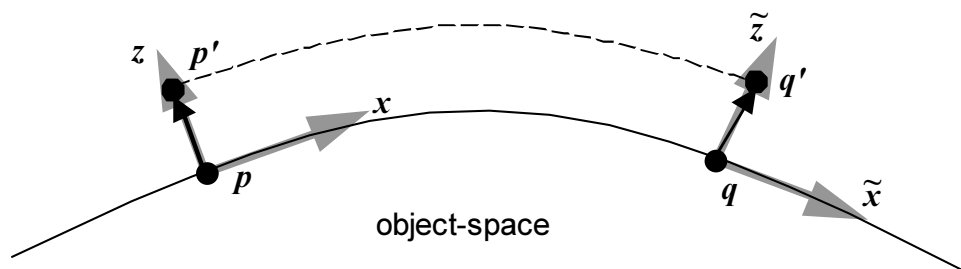


Figure 10.5 — Local tangent frame axes at reference points p and q

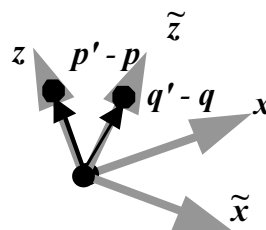


Figure 10.6 — Direction vectors in the two local tangent frames using a common origin

To support the inter-conversion of directions, vector quantities²⁸, and orientations between two SRFs, this International Standard uses the notions of reference point and local tangent frame. Since there is neither an intrinsic SRF nor an intrinsic reference point in object-space, it is necessary to specify the reference point in order to be able to inter-convert the representation of directions, vectors, or orientations between two SRFs. This method of associating reference points and local tangent frames reduces the general problem of inter-converting between two SRFs to that of inter-converting between two orthonormal linear spaces.

10.5.2 Specification of local tangent frame SRF

In this International Standard, a direction in a 3D²⁹ SRF_S is expressed as a combination of a unit vector and a reference coordinate. The unit vector is in a 3D linear orthonormal SRF, termed the local tangent frame at the reference coordinate, and is denoted by SRF_L. SRF_L is uniquely defined for each reference coordinate using the unit vectors tangent to the coordinate-component curves at the reference coordinate.

The *local tangent frame* SRF_L at a reference coordinate $c = (u_0, v_0, w_0)$ in the interior of the domain of SRF_S is specified by the SRFT [LOCOCENTRIC EUCLIDEAN 3D](#) with ORM = ORM_S and parameter values:

$$\begin{aligned} \mathbf{q} &= \mathbf{G}(u_0, v_0, w_0), \\ \mathbf{r} &= \frac{\mathbf{v}_1}{\|\mathbf{v}_1\|}, \text{ and} \\ \mathbf{s} &= \frac{\mathbf{v}_2}{\|\mathbf{v}_2\|} \end{aligned} \tag{10.12}$$

where:

$$\begin{aligned} \mathbf{v}_1 &= \left(\frac{d\mathbf{C}_1}{du} \right)_{u=u_0}, \\ \mathbf{v}_2 &= \left(\frac{d\mathbf{C}_2}{dv} \right)_{v=v_0}, \\ \mathbf{C}_1 &\text{ is the 1}^{\text{st}} \text{ coordinate-component curve at } (u_0, v_0, w_0), \text{ and} \\ \mathbf{C}_2 &\text{ is the 2}^{\text{nd}} \text{ coordinate-component curve at } (u_0, v_0, w_0). \end{aligned}$$

The vectors \mathbf{r} and \mathbf{s} are termed the *local tangent vectors* at c . Coordinate-component curves are defined in [5.5.3](#).

NOTE 1 The tangent vector to the 3rd coordinate-curve at (u_0, v_0, w_0) points in the same direction as the vector $\mathbf{t} = \mathbf{r} \times \mathbf{s}$ because of the coordinate-component ordering restriction specified in [5.6.4](#).

When SRF_S is a linear SRF, SRF_L at reference coordinate $(0, 0, 0)$ coincides with SRF_S. In addition, the unit vector that represents the direction is independent of the reference coordinate used. Linear SRFs include those based on SRFTs [CELESTIOCENTRIC](#), [LOCAL TANGENT SPACE EUCLIDEAN](#), [LOCOCENTRIC EUCLIDEAN 3D](#), and [LOCAL SPACE RECTANGULAR 3D](#).

10.5.3 Specification of direction

A *direction* in an orthogonal CS based SRF_S shall be comprised of:

- a coordinate c in the interior of the CS domain of SRF_S, and
- a unit vector \mathbf{n} in the local tangent frame at c .

²⁸ Not necessarily a direction or a unit vector, but any vector of interest.

²⁹ All of the 3D SRFTs in this International Standard are based on orthogonal CSs.

The coordinate c is termed the *reference coordinate* of the direction and its corresponding position is termed the *reference position* for the direction. The vector n is termed the *direction vector* at c .

NOTE The local tangent frame at a coordinate is an instance of the SRFT [LOCOCENTRIC EUCLIDEAN 3D](#) that provides a vector space setting for vector operations on direction vectors at c .

EXAMPLE 1 If SRF_S is a [LOCOCENTRIC EUCLIDEAN 3D](#) SRF with SRF parameters q , r and s , and c is an SRF_S reference coordinate, then local tangent vectors at c are equal to the SRF parameters r and s . If $c = (0,0,0)$, then $\text{SRF}_L = \text{SRF}_S$.

EXAMPLE 2 SRF_S is an [EQUATORIAL INERTIAL](#) SRF. This SRF is based on the [EQUATORIAL SPHERICAL](#) CS. If $c = (\lambda_0, \theta_0, \rho_0)$ is a reference coordinate, then the local tangent vectors at c are:

$$\begin{aligned} r &= \frac{v_1}{\|v_1\|} \text{ and } s = \frac{v_2}{\|v_2\|} \\ \text{where:} \\ v_1 &= \left(\frac{dC_1}{d\lambda} \right)_{\lambda=\lambda_0} = \left(\frac{d}{d\lambda} (\rho_0 \cos(\theta_0) \cos(\lambda), \rho_0 \cos(\theta_0) \sin(\lambda), \rho_0 \sin(\theta_0)) \right)_{\lambda=\lambda_0} \\ &= (-\rho_0 \cos(\theta_0) \sin(\lambda_0), \rho_0 \cos(\theta_0) \cos(\lambda_0), 0), \\ \frac{v_1}{\|v_1\|} &= (-\sin(\lambda_0), \cos(\lambda_0), 0), \\ v_2 &= \left(\frac{dC_2}{d\theta} \right)_{\theta=\theta_0} = \left(\frac{d}{d\theta} (\rho_0 \cos(\theta) \cos(\lambda_0), \rho_0 \cos(\theta) \sin(\lambda_0), \rho_0 \sin(\theta)) \right)_{\theta=\theta_0} \\ &= (-\rho_0 \sin(\theta_0) \cos(\lambda_0), -\rho_0 \sin(\theta_0) \sin(\lambda_0), \rho_0 \cos(\theta_0)), \text{ and} \\ \frac{v_2}{\|v_2\|} &= (-\sin(\theta_0) \cos(\lambda_0), -\sin(\theta_0) \sin(\lambda_0), \cos(\theta_0)). \end{aligned}$$

EXAMPLE 3 SRF_S is a [CELESTIODETIC](#) SRF. This SRF is based on the [GEODETIC](#) CS. If $c = (\lambda_0, \varphi_0, h_0)$ is a reference coordinate, then the local tangent vectors at c are:

$$\begin{aligned} r &= (-\sin(\lambda_0), \cos(\lambda_0), 0), \\ s &= (-\sin(\varphi_0) \cos(\lambda_0), -\sin(\varphi_0) \sin(\lambda_0), \cos(\varphi_0)), \text{ and} \\ t = r \times s &= (\cos \lambda_0 \cos \varphi_0, \sin \lambda_0 \cos \varphi_0, \sin \varphi_0). \end{aligned}$$

In this example, SRF_L is equivalent to a [LOCAL TANGENT SPACE EUCLIDEAN](#) SRF with template parameter values $\lambda = \lambda_0$, $\varphi = \varphi_0$, $\alpha = 0$, $x_F = y_F = 0$, and h_0 .

EXAMPLE 4 SRF_S is based on an augmented conformal map projection CS. If $c = (u_0, v_0, h_0)$ is a reference coordinate, and $(\lambda_0, \varphi_0, h_0)$ is the corresponding celestiodetic coordinate, then the local tangent vectors at c are:

$$\begin{aligned} r &= (-\sin \lambda_0 \cos \gamma_0 + \cos \lambda_0 \sin \varphi_0 \sin \gamma_0, \cos \lambda_0 \cos \gamma_0 + \sin \lambda_0 \sin \varphi_0 \sin \gamma_0, -\cos \varphi_0 \sin \gamma_0), \text{ and} \\ s &= (-\sin \lambda_0 \sin \gamma_0 - \cos \lambda_0 \sin \varphi_0 \cos \gamma_0, \cos \lambda_0 \sin \gamma_0 - \sin \lambda_0 \sin \varphi_0 \cos \gamma_0, \cos \varphi_0 \cos \gamma_0) \\ \text{where:} \\ \gamma_0 &= \gamma(\lambda_0, \varphi_0) \text{ the convergence of the meridian.} \end{aligned}$$

In this example, SRF_L is equivalent to a [LOCAL TANGENT SPACE EUCLIDEAN](#) SRF with template parameter values $\lambda = \lambda_0$, $\varphi = \varphi_0$, $\alpha = \gamma_0$, $x_F = y_F = 0$, and h_0 .

10.5.4 Changing the reference coordinate of a direction

Given a direction represented with direction vector n_1 at reference coordinate c_1 , the same direction may be represented at another reference coordinate c_2 in the same SRF, with direction vector n_2 . The direction vector n_2 is computed as:

$$\mathbf{n}_2 = \mathbf{R} \mathbf{n}_1 \quad (10.13)$$

where:

$$\mathbf{R} = \begin{pmatrix} \mathbf{r}_1 \bullet \mathbf{r}_2 & \mathbf{s}_1 \bullet \mathbf{r}_2 & \mathbf{t}_1 \bullet \mathbf{r}_2 \\ \mathbf{r}_1 \bullet \mathbf{s}_2 & \mathbf{s}_1 \bullet \mathbf{s}_2 & \mathbf{t}_1 \bullet \mathbf{s}_2 \\ \mathbf{r}_1 \bullet \mathbf{t}_2 & \mathbf{s}_1 \bullet \mathbf{t}_2 & \mathbf{t}_1 \bullet \mathbf{t}_2 \end{pmatrix},$$

\mathbf{r}_i and \mathbf{s}_i are the local tangent vectors at c_i , and

$\mathbf{t}_i = \mathbf{r}_i \times \mathbf{s}_i$ for $i = 1, 2$.

The local tangent vectors are computed as in [Equation \(10.12\)](#). The matrix \mathbf{R} in [Equation \(10.13\)](#) is the direction cosine matrix of the local tangent frame at c_2 with respect to the local tangent frame at c_1 (see [Equation \(6.6\)](#)).

If the SRF is based on a linear CS, then matrix \mathbf{R} is the identity matrix and $\mathbf{n}_1 = \mathbf{n}_2$. This implies that in a linear orthonormal SRF, a direction vector is independent of the reference coordinate. Thus, [Equation \(10.13\)](#) is only of interest in the case of a curvilinear SRF.

10.5.5 Representing a direction in a different SRF

Given a direction represented with direction vector \mathbf{n}_S at c_S in SRF_S , the same direction may be represented at reference coordinate c_T , with direction vector \mathbf{n}_T in SRF_T . If \mathbf{H}_{ST} is the similarity transformation from ORM_S to ORM_T and \mathbf{M}_{ST} is the matrix in the last term in [Equation \(10.3\)](#), then the direction vector \mathbf{n}_T is computed as:

$$\mathbf{n}_T = \mathbf{R}_{ST} \mathbf{n}_S \quad (10.14)$$

where:

$$\mathbf{R}_{ST} = \mathbf{R}_T^\top \circ \mathbf{M}_{ST} \circ \mathbf{R}_S,$$

\mathbf{R}_T^\top = the transpose of \mathbf{R}_T , and

for: $i = S$ or T ,

$$\mathbf{R}_i = \begin{pmatrix} r_{i,1} & s_{i,1} & t_{i,1} \\ r_{i,2} & s_{i,2} & t_{i,2} \\ r_{i,3} & s_{i,3} & t_{i,3} \end{pmatrix},$$

$$\mathbf{t}_i = \mathbf{r}_i \times \mathbf{s}_i = (t_{i,1} \quad t_{i,2} \quad t_{i,3}),$$

$\mathbf{r}_i = (r_{i,1} \quad r_{i,2} \quad r_{i,3})$ and $\mathbf{s}_i = (s_{i,1} \quad s_{i,2} \quad s_{i,3})$ are the local tangent vectors at c_i .

[Equation \(10.14\)](#) is derived from [Equation \(10.9\)](#) by dropping the translation term since directions are translation invariant, and dropping the scale factor $(1 + \Delta s_{SR}) / (1 + \Delta s_{TR})$ since \mathbf{n}_T is a unit vector.

The rotation matrix \mathbf{R}_{ST} in [Equation 10.14](#) is termed the *orientation of SRF_T at reference point c_T , with respect to SRF_S at reference point c_S* . The rotation matrix \mathbf{R}_{ST} is a generalization of the matrix in [Equation \(10.13\)](#) that accounts for the change of position-space between the source and target ORMs.

EXAMPLE 1 SRF_S is SRF [GEODETTIC WGS 1984](#) and SRF_T is SRF [GEOCENTRIC WGS 1984](#). With SRF_S reference coordinate $c_S = (\lambda, \varphi, h) = (-77\pi/180, +38,88\pi/180, 0)$, the Washington monument, an obelisk, points approximately in the direction $\mathbf{n}_S = (0, 0, 1)$ at c_S . In this example, $\text{ORM}_S = \text{ORM}_T$ so that \mathbf{M}_{ST} is the identity matrix, and because SRF_T is based on SRFT [CELESTIOCENTRIC](#), \mathbf{R}_T is also the identity matrix. Consequently [Equation \(10.14\)](#) reduces to:

$$\mathbf{n}_T = \mathbf{R}_S \mathbf{n}_S = \begin{pmatrix} r_{S,1} & s_{S,1} & t_{S,1} \\ r_{S,2} & s_{S,2} & t_{S,2} \\ r_{S,3} & s_{S,3} & t_{S,3} \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \mathbf{t}_S.$$

Then using the expression in [10.5.3 Example 3](#) for \mathbf{t} :

$$\begin{aligned} \mathbf{t}_S &= (\cos \lambda_0 \cos \varphi_0 \quad \sin \lambda_0 \cos \varphi_0 \quad \sin \varphi_0) \\ &= (\cos(-77\pi/180) \cos(38,88\pi/180) \quad \sin(-77\pi/180) \cos(38,88\pi/180) \quad \sin(38,88\pi/180)) \\ &= (0,175 \ 115 \ 92 \quad -0,758 \ 510 \ 36 \quad 0,627 \ 691 \ 36). \end{aligned}$$

The resulting vector $\mathbf{n}_T = (0,175 \ 115 \ 92 \quad -0,758 \ 510 \ 36 \quad 0,627 \ 691 \ 36)$ is the direction vector in SRF_T.

Engineering designs and abstract models are often intended for realization in the physical world. In such cases, the operation of changing the representation of direction vector \mathbf{n}_S in a linear SRF representing the abstract space to a direction vector \mathbf{n}_T in a linear SRF representing the physical object space is based on [Equation \(10.11\)](#). In the notation of [10.4.6](#):

$$\mathbf{n}_T = \frac{1}{|\mathbf{W}|} \mathbf{W} \circ \mathbf{R}_S \mathbf{n}_S. \quad (10.15)$$

Division by the determinant cancels any scaling by matrix \mathbf{W} to ensure that \mathbf{n}_T is a unit vector. (The rotation matrix \mathbf{R}_S does not change the length of \mathbf{n}_S .)

EXAMPLE 2 In ISO/IEC 18023-1 (see [18023-1](#)), if an instance of the class <DRM Geometry Model Instance> has a component of class <DRM World Transformation>, that component specifies an invertible matrix \mathbf{W} and a coordinate c in the <DRM Environment Root> SRF. If c_S and \mathbf{n}_S are a reference coordinate and a direction vector in an associated [LOCAL SPACE RECTANGULAR 3D](#) <DRM Geometry Model>, and SRF_T is the local tangent frame at c , then [Equation \(10.11\)](#) and [Equation \(10.15\)](#) may be used to compute c_T and \mathbf{n}_T , respectively. The methods of [10.4.3](#) may be used to further change c_T from SRF_T to the <DRM Environment Root> SRF. This procedure to change <DRM Geometry Model> coordinates and directions to the environment root SRF is termed "model instancing".

10.5.6 Representing a vector quantity in a different SRF

Vectors combine a direction with a magnitude, and are used to describe a number of properties of moving objects, such as their velocities, accelerations and other quantities. Similar to directions, vectors are defined with respect to the axes of a specific SRF_L at a reference point. All properties and operations that apply to directions also apply to vectors.

Given a vector quantity \mathbf{v}_S at reference location c_S in SRF_S, it may be represented as \mathbf{v}_T at reference location c_T in SRF_T. If \mathbf{R}_{ST} is the orientation SRF_S at reference point c_S , with respect to SRF_T at reference point c_T , the vector \mathbf{v}_T is computed as:

$$\mathbf{v}_T = \mathbf{R}_{ST} \mathbf{v}_S$$

Given a vector quantity \mathbf{v}_B in a body frame SRF_B (or in general any linear orthonormal reference frame) whose orientation at a reference location c_S in SRF_S is known, that vector quantity may be represented as \mathbf{v}_T at reference location c_T in SRF_T. If \mathbf{R}_{BS} is the orientation of SRF_B at reference point c_S , with respect to SRF_S, the vector \mathbf{v}_T is computed as: $\mathbf{v}_T = \mathbf{R}_{ST} \circ \mathbf{R}_{BS} \mathbf{v}_B$.

10.5.7 Representing an orientation in a different SRF

The orientation of an object E in 3D space specifies how a set of orthogonal axes attached to that object are aligned with respect to the axes of a specific orthogonal SRF (see [6.1](#)). The orientation of an object specifies a rotation operation that would bring the SRF axes into alignment with the corresponding object axes (or vice versa).

As with directions, orientations that are specified with respect to an SRF use the unique local tangent frame SRF_L at a specified reference location (see [10.5.2](#)).

Given R_{ES} , the orientation of an object E at reference location c_S with respect to SRF_S , the orientation of that object may be represented as R_{ET} , at reference location c_T , with respect to SRF_T . If R_{ST} is the orientation of the SRF_S at reference point c_S , with respect to SRF_T at reference point c_T , the orientation R_{ET} is computed as: $R_{ET} = R_{ST} \circ R_{ES}$ (see [Equation \(6.2\)](#)).

10.6 Euclidean distance

This International Standard supports an operation to return the Euclidean distance between two object-space locations using the coordinates of those locations in an SRF.

If c_1 and c_2 are two coordinates in an SRF, and if G is the generating function of the CS of the SRF, the *Euclidean distance* d_E between the corresponding points in object-space is given by:

$$d_E(c_1, c_2) = d(G(c_1), G(c_2))$$

where d is the [Euclidean metric](#).

10.7 Geodesic distance operations

10.7.1 Introduction

A curve on a smooth surface that has the property that any sufficiently small segment of it realizes the shortest distance on the surface between the segment's two endpoints is called a geodesic. The formal definition of a geodesic is given in [A.7.4](#).

EXAMPLE 1 On a sphere, the equator, the meridians, and all other great circles are geodesics. Likewise any segment of one of these curves is a geodesic. No parallel of latitude except the equator is a geodesic.

EXAMPLE 2 On an oblate ellipsoid, the equator is a geodesic, and the meridians are all geodesics. All the other geodesics are curves which cross the equator at some non-right angle and wind around the ellipsoid between two parallels of opposite latitude (see [Figure 10.7](#)).

Let points p_1 and p_2 lie on a smooth surface. The shortest distance on the surface from p_1 to p_2 is the shortest arc length associated with any of the smooth surface curves that connect p_1 to p_2 . This distance is unique, but the curve that has this arc length may not be unique. In particular, for the two pole points, every meridian is such a curve.

EXAMPLE 3 On an oblate ellipsoid, let p_1 be the point with surface geodetic coordinates $(\lambda, \varphi) = (0^\circ, 20^\circ)$ and let p_2 be the point diametrically opposite, *i.e.*, with surface geodetic coordinates $(\lambda, \varphi) = (180^\circ, -20^\circ)$. Then the shortest distance on the surface from p_1 to p_2 is twice the meridional quadrant, *i.e.*, twice the length of a meridian from equator to pole. But there are two distinct curves from p_1 to p_2 which have this number as their arc length – one passes through the north pole and the other passes through the south pole. (Both are composed of segments of meridians).

EXAMPLE 4 On an oblate ellipsoid with eccentricity ε , let points p_1 and p_2 lie on the equator but be separated by a longitude difference that is less than π and more than $\pi\sqrt{1-\varepsilon^2}$, an angle called the “lift-off longitude”. Then there will be two curves from p_1 to p_2 whose arc length is the shortest distance from p_1 to p_2 – one lying in the northern hemisphere, the other lying (symmetrically) in the southern hemisphere.

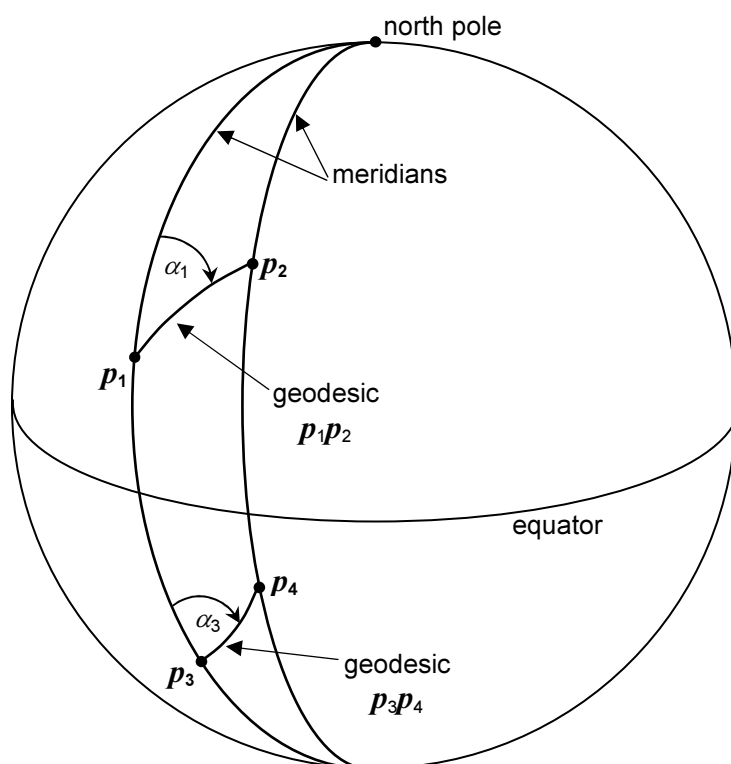


Figure 10.7 — Examples of geodesics

If a curve lying on a smooth surface connects point p_1 to point p_2 , and if that curve's arc length is also the shortest distance from p_1 to p_2 , then that curve is a geodesic. Thus, the arc length of the shortest curve connecting the two points is termed the *geodesic distance*.

EXAMPLE 5 The two curves from p_1 to p_2 defined in Example 3 are geodesics.

EXAMPLE 6 The two curves from p_1 to p_2 defined in Example 4 are geodesics.

The converse is not true. If a geodesic starts at point p_1 and ends at point p_2 , its arc length may or may not be the same as the shortest distance on the surface from p_1 to p_2 .

EXAMPLE 7 Let points p_1 and p_2 lie on the equator of a sphere or oblate ellipsoid at longitudes 0° and 181° , respectively. The segment of the equator from p_1 to p_2 that is continuous in longitude from 0° to 181° is a geodesic. (All segments of the equator are geodesics). However, its arc length is not the shortest distance on the surface from p_1 to p_2 . Any curve which realizes the shortest distance on the surface from p_1 to p_2 has to lie within a single hemisphere of longitude.

There are two problems of interest pertaining to geodesics on an oblate ellipsoid. In the first, called the direct problem, a surface point, an azimuth, and a distance are given. The problem is to find a second surface point which terminates the (unique) geodesic whose initial point is the given point, whose initial forward azimuth is the given azimuth, and whose arc length is the given distance. Also to be found is the geodesic's terminal forward azimuth. The details are given in [10.7.3](#).

In the second problem, called the indirect problem, two distinct surface points are given. The problem is to find the shortest distance on the surface between the two given points, and find the set of curves (which will be geodesics) whose arc lengths equal this shortest distance. In addition, the initial and terminal forward azimuths of each curve is to be found. The details are in [10.7.4](#).

This International Standard supports the geodesic operations for SRFs based on SRFT [CELESTIODETC](#), [PLANETODETC](#), and all map projection SRFTs.

Given two surface coordinates c_1 and c_2 of points p_1 and p_2 , respectively, the *geodesic distance operation*:

$$s = d_G(c_1, c_2)$$

is defined as the indirect problem for (λ_1, φ_1) and (λ_2, φ_2) where (λ_1, φ_1) is the surface geodetic coordinate for c_1 and (λ_2, φ_2) is the surface geodetic coordinate for c_2 .

An extended version of this operation provides the forward azimuth value α_1 at c_1 and the forward azimuth value α_2 at c_2 :

$$\{s, \alpha_1, \alpha_2\} = d_{GI}(c_1, c_2).$$

The geodesic destination operation requires a starting point c_1 , a forward azimuth value α_1 , at c_1 and a positive distance s . It returns the destination point c_2 and the forward azimuth value α_2 at c_2 :

$$\{c_2, \alpha_2\} = d_{GD}(c_1, \alpha_1, s)$$

where $\{(\lambda_2, \varphi_2), \alpha_2\}$ is the direct problem solution for input parameter values $\{(\lambda_1, \varphi_1), \alpha_1, s\}$.

There is a large body of literature concerning computational techniques to solve the direct and indirect problems. In the interest of accuracy and computational efficiency, many of these computational techniques treat the problems by sub-cases -- short lines, long lines, intermediate length lines, and other caveats and exceptions. Each of these has been optimized in a way that is appropriate for the intended application or user domain. For purposes of this International Standard, a recently published treatment ([\[ROL10\]](#)) that has one mathematical formulation to cover all cases is utilized.

10.7.2 Auxiliary functions

The treatment of the direct and indirect problems in [10.7.3](#) and [10.7.4](#) require the auxiliary functions defined in this subclause.

An important characteristic of a geodesic on an oblate ellipsoid is that the quantity termed the (non-metric) *Clairaut constant* and defined by:

$$c = \frac{\sin(\alpha)\cos(\varphi)}{\sqrt{1-\varepsilon^2\sin^2(\varphi)}}$$

has a constant value at every point on a given geodesic, where (λ, φ) is the coordinate of a point on the geodesic and α is the azimuth of the curve at that point.

The mathematics required to solve the direct and indirect problems involves the use of elliptic integrals. The incomplete elliptic integral of third kind is defined for real n, θ , and m , with $m^2 < 1$ as:

$$P(n, \theta, m) = \int_0^\theta \frac{d\xi}{(1-n\sin^2\xi)\sqrt{1-m\sin^2\xi}}.$$

The treatment in [\[ROL10\]](#) defines two auxiliary functions: a longitude difference function $L(c, \theta_1, \theta_2)$ and an arc length function $A(c, \theta_1, \theta_2)$ that are defined for all values of c, θ_1 and θ_2 by:

$$L(c, \theta_1, \theta_2) = \frac{c(1-\varepsilon^2)}{\sqrt{1-c^2\varepsilon^2}} \left(P(k^2, \theta_2, k^2\varepsilon^2) - P(k^2, \theta_1, k^2\varepsilon^2) \right), \quad c \neq 0 \quad (10.16)$$

$$L(0, \theta_1, \theta_2) = \lim_{c \rightarrow 0^+} L(c, \theta_1, \theta_2)$$

and

$$A(c, \theta_1, \theta_2) = \frac{a(1-\varepsilon^2)}{\sqrt{1-c^2\varepsilon^2}} \left(P(k^2\varepsilon^2, \theta_2, k^2\varepsilon^2) - P(k^2\varepsilon^2, \theta_1, k^2\varepsilon^2) \right). \quad (10.17)$$

where

$$k^2 = \frac{1-c^2}{1-c^2\varepsilon^2}.$$

10.7.3 The direct problem

Given an oblate ellipsoid with major semi-axis a and eccentricity ε , let p_1 be a non-polar point on the ellipsoid given by its surface geodetic coordinates (λ_1, φ_1) . Let a geodesic be defined with p_1 as its initial point, α_1 as its initial forward azimuth, and arc length s . This geodesic will terminate at a point p_2 .

The direct problem requires finding the surface geodetic coordinates (λ_2, φ_2) of p_2 and the forward azimuth α_2 of the geodesic at the point p_2 . The quantity $\alpha_2 + \pi$ is termed the *back azimuth* at p_2 as it points backwards toward p_1 .

The given parameters are restricted to $-\pi/2 < \varphi_1 < \pi/2$, $-\pi < \alpha_1 \leq \pi$, and $s > 0$.

The functions $L(c, \theta_1, \theta_2)$ and $A(c, \theta_1, \theta_2)$ are used to solve the direct problem.

The given values in the direct problem (λ_1, φ_1) and α_1 determine c ,

$$c = \frac{\sin(\alpha_1) \cos(\varphi_1)}{\sqrt{1-\varepsilon^2 \sin^2(\varphi_1)}}.$$

Then,

$$\begin{aligned} \lambda_2 &= \lambda_1 + L(c, \theta_1, \theta_2), \\ \varphi_2 &= \arcsin(k \sin \theta_2), \text{ and} \\ \alpha_2 &= \arctan 2 \left(c \sqrt{1-k^2 \sin \theta_2}, k \sqrt{1-k^2 \varepsilon^2 \sin \theta_2} \right) \end{aligned}$$

where

$$\theta_1 = \arcsin(\sin(\varphi_1)/k), \quad k = \pm \sqrt{\frac{1-c^2}{1-c^2\varepsilon^2}}, \quad k \geq 0 \text{ if } |\alpha_1| \leq \frac{\pi}{2} \text{ and } k < 0 \text{ otherwise, and}$$

θ_2 is determined by:

$$s = A(c, \theta_1, \theta_2). \quad (10.18)$$

Equation 10.18 has a unique solution for θ_2 . Reference [ROL12] gives the following Newton-Raphson iteration, which rapidly converges to the solution:

$$\theta_2^{[1]} = \theta_1 + \frac{\pi s}{A(c, 0, \pi)},$$

$$\theta_2^{[n+1]} = \theta_2^{[n]} - \left(\frac{\sqrt{1-c^2\varepsilon^2}}{a(1-\varepsilon^2)} \right) \left(1 - k^2\varepsilon^2 \sin \theta_2^{[n]} \right)^{3/2} \left(A(c, \theta_1, \theta_2^{[n]}) - s \right).$$

10.7.4 The indirect problem

Given an oblate ellipsoid with major semi-axis a and eccentricity ε , let p_1 and p_2 be two points on the ellipsoid given by their surface geodetic coordinates (λ_1, φ_1) and (λ_2, φ_2) .

The indirect problem requires finding the shortest distance s on the ellipsoid from p_1 to p_2 . Further, for each curve from p_1 to p_2 whose arc length is s , it is required to find the forward azimuths α_1 and α_2 at the points p_1 and p_2 respectively. (Such curves will be geodesics, and there will be 1, 2, or infinitely many of them.)

The given parameters are restricted to $-\pi \leq \lambda_2 - \lambda_1 \leq \pi$, $-\pi/2 \leq \varphi_1 \leq \pi/2$, and $-\pi/2 \leq \varphi_2 \leq \pi/2$.

The solution to the indirect problem can be determined once c , the Clairaut constant for the solution geodesic curve segment, is found. Dealing with the extreme c values 0 and 1 separately simplifies the process.

The single meridional case: $c = 0$ if $\lambda_2 = \lambda_1$ or if either point is a pole ($|\varphi_1| = \pi/2$ or $|\varphi_2| = \pi/2$). Then if $\varphi_1 < \varphi_2$, the solution is:

$$s = A(0, \varphi_1, \varphi_2), \text{ and } \alpha_1 = \alpha_2 = 0.$$

Otherwise $\varphi_1 > \varphi_2$, and the solution is:

$$s = A(0, \varphi_2, \varphi_1), \text{ and } \alpha_1 = \alpha_2 = \pi.$$

If either point is a pole, the azimuth at that point is undefined. The solution geodesic curve segment is unique unless both given points are poles. In that case the solution set is the infinite set of all meridians.

Meridional segments joined at pole: $c = 0$ if $\lambda_2 = \lambda_1 \pm \pi$ and $\varphi_2 \geq -\varphi_1$. Then

$$s = A(0, \varphi_1, \pi - \varphi_2), \alpha_1 = 0, \alpha_2 = \pi$$

and the geodesic curve segment passes through the north pole.

Similarly, $c = 0$ if $\lambda_2 = \lambda_1 \pm \pi$ and $\varphi_2 < -\varphi_1$. Then

$$s = A(0, \varphi_1, -\varphi_2 - \pi), \alpha_1 = \pi, \alpha_2 = 0 \text{ and the geodesic curve segment passes through the south pole.}$$

Equatorial segment: $c = 1$ if $\varphi_1 = \varphi_2 = 0$ and $0 < |\lambda_2 - \lambda_1| \leq \pi\sqrt{1-\varepsilon^2}$. Then

$$s = a|\lambda_2 - \lambda_1|, \text{ and } \alpha_1 = \alpha_2 = \frac{\pi}{2} \text{ if } \alpha_1 < \alpha_2 \text{ and } \alpha_1 = \alpha_2 = -\frac{\pi}{2} \text{ otherwise.}$$

The solution is unique.

Near-antipodal equatorial segment: If $\varphi_1 = \varphi_2 = 0$ and the points are separated by more than the lift-off longitude ($\pi\sqrt{1-\varepsilon^2} < |\lambda_2 - \lambda_1| < \pi$), then c is determined by solving the equation:

$$\lambda_2 - \lambda_1 = L(c, 0, \pi) \text{ in the interval } 0 \leq c \leq 1.$$

Assuming $\lambda_1 < \lambda_2$, the solution parameters are then given by:

$$s = A(c, 0, \pi), \alpha_1 = \arcsin(c), \text{ and } \alpha_2 = \pi - \alpha_1.$$

This geodesic curve segment lies in the northern hemisphere. A second solution lies in the southern hemisphere in north-south symmetry.

Prograde typical: The remaining cases may be reduced to the prograde typical case of $0 < \lambda_2 - \lambda_1 < \pi$ and $\varphi_2 \geq |\varphi_1| \neq 0$.

$$\text{Define } c_{\max} = \frac{\cos \varphi_2}{\sqrt{1-\varepsilon^2} \sin^2 \varphi_2} \text{ and } \lambda_{\text{crit}} = L\left(c_{\max}, \left(\frac{\sin \varphi_1}{\sin \varphi_2}\right), \frac{\pi}{2}\right).$$

Then c may be determined by an iterative solution of the equation:

$$\lambda_2 - \lambda_1 = L(c, \theta_1(c), \theta_2(c)) \text{ in the interval } 0 \leq c \leq c_{\max},$$

where

$$\theta_1(c) = \arcsin(\sin(\varphi_1)/k(c)), \quad k(c) = \sqrt{\frac{1-c^2}{1-c^2\varepsilon^2}}, \text{ and}$$

$$\theta_1(c) = \begin{cases} \arcsin(\sin(\varphi_1)/k(c)), & \text{if } \lambda_2 - \lambda_1 < \lambda_{\text{crit}} \\ \pi/2, & \text{if } \lambda_2 - \lambda_1 = \lambda_{\text{crit}} \\ \pi - \arcsin(\sin(\varphi_1)/k(c)), & \text{if } \lambda_2 - \lambda_1 > \lambda_{\text{crit}} \end{cases}.$$

The solution parameters are determined by c :

$$s = A(c, \theta_1(c), \theta_2(c)),$$

$$\alpha_1 = \arctan 2\left(c\sqrt{1-\varepsilon^2} \sin \varphi_1, k(c)\sqrt{1-c^2} \cos \varphi_1\right), \text{ and}$$

$$\alpha_2 = \arctan 2\left(c\sqrt{1-\varepsilon^2} \sin \varphi_2, k(c)\sqrt{1-c^2} \cos \varphi_2\right).$$

NOTE: Extremely small values of c can cause numerical instability in some implementations. Alternative methods to evaluate $L(c, \theta_1, \theta_2)$ in this and other difficult cases are treated in [\[ROL12\]](#).

Other prograde cases: If $0 < \lambda_2 - \lambda_1 < \pi$ and cases above do not apply, a new pair of points p_3 and p_4 that satisfy the prototypical case constraints can be specified using parameters from the given pair p_1 and p_2 . The indirect problem solution for points p_3 and p_4 , the shortest distance between them \tilde{s} , and the forward azimuths α_3 and α_4 will determine the solution for p_1 and p_2 as follows:

If $|\varphi_2| \leq \varphi_1$, let $p_3 = (\lambda_1, \varphi_2)$ and $p_4 = (\lambda_2, \varphi_1)$. Then $\alpha_1 = \pi - \alpha_4$ and $\alpha_2 = \pi - \alpha_3$.

If $|\varphi_2| \leq -\varphi_1$, let $p_3 = (\lambda_1, -\varphi_2)$ and $p_4 = (\lambda_2, -\varphi_1)$. Then $\alpha_1 = \alpha_4$ and $\alpha_2 = \alpha_3$.

If $|\varphi_1| \leq -\varphi_2$, let $p_3 = (\lambda_1, -\varphi_1)$ and $p_4 = (\lambda_2, -\varphi_2)$. Then $\alpha_1 = \pi - \alpha_3$ and $\alpha_2 = \pi - \alpha_4$.

In all these cases the arc length solution is the same, $s = \tilde{s}$, and the value of c and the multiplicity of shortest geodesic segments are also the same.

Retrograde cases: A retrograde case, $\lambda_2 < \lambda_1$, is converted to a prograde case with $p_3 = (\lambda_2, \varphi_1)$ and $p_4 = (\lambda_1, \varphi_2)$. Then $\alpha_1 = -\alpha_3$, $\alpha_2 = -\alpha_4$, and $s = \tilde{s}$. The value $-c$ from prograde case is the retrograde solution value for c and the multiplicity of shortest geodesic segments are the same.

<http://standards.iso.org/ittf/PubliclyAvailableStandards/>