Applications of Neutrosophic Logic to Robotics

An Introduction

Florentin Smarandache
University of New Mexico
Gallup, NM 87301, USA
E-mail: smarand@unm.edu

Luige Vlădăreanu
Romanian Academy, Institute of Solid Mechanics
15 C-tin Mille, 010141 Bucharest 1, Romania
E-mail: luigiv@arexim.ro

Abstract— In this paper we present the N-norms/N-conorms in neutrosophic logic and set as extensions of T-norms/T-conorms in fuzzy logic and set.
Then we show some applications of the neutrosophic logic to robotics.

Keywords: N-norm, N-conorm, N-pseudonorm, N-pseudoconorm, Neutrosophic set, Neutrosophic logic, Robotics

I. DEFINITION OF NEUTROSOPHIC SET

Let T, I, F be real standard or non-standard subsets of $]^-0, 1^+[$,
with sup T = t_sup, inf T = t_inf,
sup I = i_sup, inf I = i_inf,
sup F = f_sup, inf F = f_inf,
and n_sup = t_sup+i_sup+f_sup,
n_inf = t_inf+i_inf+f_inf.
Let U be a universe of discourse, and M a set included in U. An element x from U is noted with respect to the set M as x(T, I, F) and belongs to M in the following way: it is t% true in the set, i% indeterminate (unknown if it is or not) in the set, and f% false, where t varies in T, i varies in I, f varies in F ([1], [3]).
Statically T, I, F are subsets, but dynamically T, I, F are functions/operators depending on many known or unknown parameters.

II. DEFINITION OF NEUTROSOPHIC LOGIC

In a similar way we define the **Neutrosophic Logic**:
A logic in which each proposition x is T% true, I% indeterminate, and F% false, and we write it x(T,I,F), where T, I, F are defined above.

III. PARTIAL ORDER

We define a *partial order relationship* on the neutrosophic set/logic in the following way:

$x(T_1, I_1, F_1) \leq y(T_2, I_2, F_2)$ iff (if and only if)
$T_1 \leq T_2, I_1 \geq I_2, F_1 \geq F_2$ for crisp components.

And, in general, for subunitary set components:

$x(T_1, I_1, F_1) \leq y(T_2, I_2, F_2)$ iff
$\inf T_1 \leq \inf T_2, \sup T_1 \leq \sup T_2,$
$\inf I_1 \geq \inf I_2, \sup I_1 \geq \sup I_2,$
$\inf F_1 \geq \inf F_2, \sup F_1 \geq \sup F_2.$

If we have mixed - crisp and subunitary - components, or only crisp components, we can transform any crisp component, say "a" with a Ï [0,1] or a Ï $]^-0, 1^+[$, into a subunitary set [a, a]. So, the definitions for subunitary set components should work in any case.

IV. N-NORM AND N-CONORM

As a generalization of T-norm and T-conorm from the Fuzzy Logic and Set, we now introduce the **N-norms and N-conorms** for the Neutrosophic Logic and Set.

A. *N-norm*

N_n: $( ]^-0,1^+[\times]^-0,1^+[\times]^-0,1^+[ )^2 \rightarrow]^-0,1^+[\times]^-0,1^+[\times]^-0,1^+[$
$N_n (x(T_1,I_1,F_1), y(T_2,I_2,F_2)) = (N_nT(x,y), N_nI(x,y), N_nF(x,y))$, where $N_nT(.,.)$, $N_nI(.,.)$, $N_nF(.,.)$ are the truth/membership, indeterminacy, and respectively falsehood/nonmembership components.

N_n have to satisfy, for any x, y, z in the neutrosophic logic/set M of the universe of discourse U, the following axioms:
a) Boundary Conditions: $N_n(x, \mathbf{0}) = \mathbf{0}$, $N_n(x, \mathbf{1}) = x$.
b) Commutativity: $N_n(x, y) = N_n(y, x)$.
c) Monotonicity: If $x \leq y$, then $N_n(x, z) \leq N_n(y, z)$.
d) Associativity: $N_n(N_n (x, y), z) = N_n(x, N_n(y, z))$.

There are cases when not all these axioms are satisfied, for example the associativity when dealing with the neutrosophic normalization after each neutrosophic operation. But, since we work with approximations, we can call these **N-pseudo-norms**, which still give good results in practice.

N_n represent the *and* operator in neutrosophic logic, and respectively the *intersection* operator in neutrosophic set theory.

Let J ∈ {T, I, F} be a component.
Most known N-norms, as in fuzzy logic and set the T-norms, are:
• The Algebraic Product N-norm: $N_{n-algebraic}J(x, y) = x \cdot y$
• The Bounded N-Norm: $N_{n-bounded}J(x, y) = \max\{0, x + y - 1\}$
• The Default (min) N-norm: $N_{n-min}J(x, y) = \min\{x, y\}$.

A general example of N-norm would be this.
Let $x(T_1, I_1, F_1)$ and $y(T_2, I_2, F_2)$ be in the neutrosophic set/logic M. Then:

$$\mathbf{N_n(x, y) = (T_1 \wedge T_2, I_1 \vee I_2, F_1 \vee F_2)}$$

where the "$\wedge$" operator, acting on two (standard or non-standard) subunitary sets, is a N-norm (verifying the above N-norms axioms); while the "$\vee$" operator, also acting on two (standard or non-standard) subunitary sets, is a N-conorm (verifying the below N-conorms axioms).

For example, $\wedge$ can be the Algebraic Product T-norm/N-norm, so $T_1 \wedge T_2 = T_1 \cdot T_2$ (herein we have a product of two subunitary sets – using simplified notation); and $\vee$ can be the Algebraic Product T-conorm/N-conorm, so $T_1 \vee T_2 = T_1+T_2-T_1 \cdot T_2$ (herein we have a sum, then a product, and afterwards a subtraction of two subunitary sets).

Or $\wedge$ can be any T-norm/N-norm, and $\vee$ any T-conorm/N-conorm from the above and below; for example the easiest way would be to consider the *min* for crisp components (or *inf* for subset components) and respectively *max* for crisp components (or *sup* for subset components).

If we have crisp numbers, we can at the end neutrosophically normalize.

B. *N-conorm*

N_c: $( ]^-0,1^+[\times]^-0,1^+[\times]^-0,1^+[ )^2 \to]^-0,1^+[\times]^-0,1^+[\times]^-0,1^+[$
$N_c (x(T_1,I_1,F_1), y(T_2,I_2,F_2)) = (N_cT(x,y), N_cI(x,y), N_cF(x,y))$, where $N_nT(.,.)$, $N_nI(.,.)$, $N_nF(.,.)$ are the truth/membership, indeterminacy, and respectively falsehood/nonmembership components.

N_c have to satisfy, for any x, y, z in the neutrosophic logic/set M of universe of discourse U, the following axioms:
a) Boundary Conditions: $N_c(x, \mathbf{1}) = \mathbf{1}$, $N_c(x, \mathbf{0}) = x$.
b) Commutativity: $N_c (x, y) = N_c(y, x)$.
c) Monotonicity: if $x \le y$, then $N_c(x, z) \le N_c(y, z)$.
d) Associativity: $N_c (N_c(x, y), z) = N_c(x, N_c(y, z))$.

There are cases when not all these axioms are satisfied, for example the associativity when dealing with the neutrosophic normalization after each neutrosophic operation. But, since we work with approximations, we can call these **N-pseudo-conorms**, which still give good results in practice.

N_c represent the *or* operator in neutrosophic logic, and respectively the *union* operator in neutrosophic set theory.

Let $J \in \{T, I, F\}$ be a component.
Most known N-conorms, as in fuzzy logic and set the T-conorms, are:
• The Algebraic Product N-conorm: $N_{c-algebraic}J(x, y) = x + y - x \cdot y$
• The Bounded N-conorm: $N_{c-bounded}J(x, y) = \min\{1, x + y\}$
• The Default (max) N-conorm: $N_{c-max}J(x, y) = \max\{x, y\}$.

A general example of N-conorm would be this.
Let $x(T_1, I_1, F_1)$ and $y(T_2, I_2, F_2)$ be in the neutrosophic set/logic M. Then:

$$\mathbf{N_n(x, y) = (T_1 \vee T_2, I_1 \wedge I_2, F_1 \wedge F_2)}$$

Where – as above - the "$\wedge$" operator, acting on two (standard or non-standard) subunitary sets, is a N-norm (verifying the above N-norms axioms); while the "$\vee$" operator, also acting on two (standard or non-standard) subunitary sets, is a N-conorm (verifying the above N-conorms axioms).

For example, $\wedge$ can be the Algebraic Product T-norm/N-norm, so $T_1 \wedge T_2 = T_1 \cdot T_2$ (herein we have a product of two subunitary sets); and $\vee$ can be the Algebraic Product T-conorm/N-conorm, so $T_1 \vee T_2 = T_1+T_2-T_1 \cdot T_2$ (herein we have a sum, then a product, and afterwards a subtraction of two subunitary sets).

Or $\wedge$ can be any T-norm/N-norm, and $\vee$ any T-conorm/N-conorm from the above; for example the easiest way would be to consider the *min* for crisp components (or *inf* for subset components) and respectively *max* for crisp components (or *sup* for subset components).

If we have crisp numbers, we can at the end neutrosophically normalize.

Since the min/max (or inf/sup) operators work the best for subunitary set components, let's present their definitions below. They are extensions from subunitary intervals {defined in [3]} to any subunitary sets. Analogously we can do for all neutrosophic operators defined in [3].

Let $x(T_1, I_1, F_1)$ and $y(T_2, I_2, F_2)$ be in the neutrosophic set/logic M.

C. *More Neutrosophic Operators*

Neutrosophic Conjunction/Intersection:

$x \wedge y=(T_\wedge,I_\wedge,F_\wedge)$,
where $\inf T_\wedge = \min\{\inf T_1, \inf T_2\}$
$\sup T_\wedge = \min\{\sup T_1, \sup T_2\}$
$\inf I_\wedge = \max\{\inf I_1, \inf I_2\}$
$\sup I_\wedge = \max\{\sup I_1, \sup I_2\}$
$\inf F_\wedge = \max\{\inf F_1, \inf F_2\}$
$\sup F_\wedge = \max\{\sup F_1, \sup F_2\}$

Neutrosophic Disjunction/Union:

$x \vee y=(T_\vee,I_\vee,F_\vee)$,
where $\inf T_\vee = \max\{\inf T_1, \inf T_2\}$
$\sup T_\vee = \max\{\sup T_1, \sup T_2\}$
$\inf I_\vee = \min\{\inf I_1, \inf I_2\}$
$\sup I_\vee = \min\{\sup I_1, \sup I_2\}$
$\inf F_\vee = \min\{\inf F_1, \inf F_2\}$
$\sup F_\vee = \min\{\sup F_1, \sup F_2\}$

Neutrosophic Negation/Complement:

$C(x) = (T_C,I_C,F_C)$,
where $T_C = F_1$
$\inf I_C = 1-\sup I_1$

$$\sup I_C = 1\text{-}\inf I_1$$
$$F_C = T_1$$

Upon the above Neutrosophic Conjunction/Intersection, we can define the

Neutrosophic Containment:

We say that the neutrosophic set A is included in the neutrosophic set B of the universe of discourse U, iff for any $x(T_A, I_A, F_A)$ Ï A with $x(T_B, I_B, F_B)$ Ï B we have:

$\inf T_A \leq \inf T_B$; $\sup T_A \leq \sup T_B$;
$\inf I_A \geq \inf I_B$; $\sup I_A \geq \sup I_B$;
$\inf F_A \geq \inf F_B$; $\sup F_A \geq \sup F_B$.

D. Remarks

a) The non-standard unit interval $]^-0, 1^+[$ is merely used for philosophical applications, especially when we want to make a distinction between relative truth (truth in at least one world) and absolute truth (truth in all possible worlds), and similarly for distinction between relative or absolute falsehood, and between relative or absolute indeterminacy.

But, for technical applications of neutrosophic logic and set, the domain of definition and range of the N-norm and N-conorm can be restrained to the normal standard real unit interval [0, 1], which is easier to use, therefore:

$$N_n: ([0,1] \times [0,1] \times [0,1])^2 \rightarrow [0,1] \times [0,1] \times [0,1]$$
and
$$N_c: ([0,1] \times [0,1] \times [0,1])^2 \rightarrow [0,1] \times [0,1] \times [0,1].$$

b) Since in NL and NS the sum of the components (in the case when T, I, F are crisp numbers, not sets) is not necessary equal to 1 (so the normalization is not required), we can keep the final result un-normalized.
But, if the normalization is needed for special applications, we can normalize at the end by dividing each component by the sum all components.
If we work with intuitionistic logic/set (when the information is incomplete, i.e. the sum of the crisp components is less than 1, i.e. *sub-normalized*), or with paraconsistent logic/set (when the information overlaps and it is contradictory, i.e. the sum of crisp components is greater than 1, i.e. *over-normalized*), we need to define the neutrosophic measure of a proposition/set.
If x(T,I,F) is a NL/NS, and T,I,F are crisp numbers in [0,1], then the **neutrosophic vector norm** of variable/set x is the sum of its components:
$$N_{vector\text{-}norm}(x) = T+I+F.$$

Now, if we apply the N_n and N_c to two propositions/sets which maybe intuitionistic or paraconsistent or normalized (i.e. the sum of components less than 1, bigger than 1, or equal to 1), x and y, what should be the neutrosophic measure of the results $N_n(x,y)$ and $N_c(x,y)$?
Herein again we have more possibilities:

- either the product of neutrosophic measures of x and y:
  $N_{vector\text{-}norm}(N_n(x,y)) = N_{vector\text{-}norm}(x)\cdot N_{vector\text{-}norm}(y)$,
- or their average:
  $N_{vector\text{-}norm}(N_n(x,y)) = (N_{vector\text{-}norm}(x) + N_{vector\text{-}norm}(y))/2$,
- or other function of the initial neutrosophic measures:

$N_{vector\text{-}norm}(N_n(x,y)) = f(N_{vector\text{-}norm}(x), N_{vector\text{-}norm}(y))$, where f(.,.) is a function to be determined according to each application.

Similarly for $N_{vector\text{-}norm}(N_c(x,y))$.
Depending on the adopted neutrosophic vector norm, after applying each neutrosophic operator the result is neutrosophically normalized. We'd like to mention that "**neutrosophically normalizing**" doesn't mean that the sum of the resulting crisp components should be 1 as in fuzzy logic/set or intuitionistic fuzzy logic/set, but the sum of the components should be as above: either equal to the product of neutrosophic vector norms of the initial propositions/sets, or equal to the neutrosophic average of the initial propositions/sets vector norms, etc.
In conclusion, we neutrosophically normalize the resulting crisp components T\`,I\`,F\` by multiplying each neutrosophic component T\`,I\`,F\` with S/(T\`+I\`+F\`), where
$S= N_{vector\text{-}norm}(N_n(x,y))$ for a N-norm or $S= N_{vector\text{-}norm}(N_c(x,y))$ for a N-conorm - as defined above.

c) If T, I, F are subsets of [0, 1] the problem of neutrosophic normalization is more difficult.
   i) If sup(T)+sup(I)+sup(F) < 1, we have an *intuitionistic proposition/set*.
   ii) If inf(T)+inf(I)+inf(F) > 1, we have a *paraconsistent proposition/set*.
   iii) If there exist the crisp numbers $t \in T$, $i \in I$, and $f \in F$ such that t+i+f =1, then we can say that we have a *plausible normalized proposition/set*.
   But in many such cases, besides the normalized particular case showed herein, we also have crisp numbers, say $t_1 \in T$, $i_1 \in I$, and $f_1 \in F$ such that $t_1+i_1+f_1 < 1$ (incomplete

information) and $t_2 \in T$, $i_2 \in I$, and $f_2 \in F$ such that $t_2+i_2+f_2 > 1$ (paraconsistent information).

E. *Examples of Neutrosophic Operators which are N-norms or N-pseudonorms or, respectively N-conorms or N-pseudoconorms*

We define a binary **neutrosophic conjunction (intersection)** operator, which is a particular case of a N-norm (neutrosophic norm, a generalization of the fuzzy T-norm):

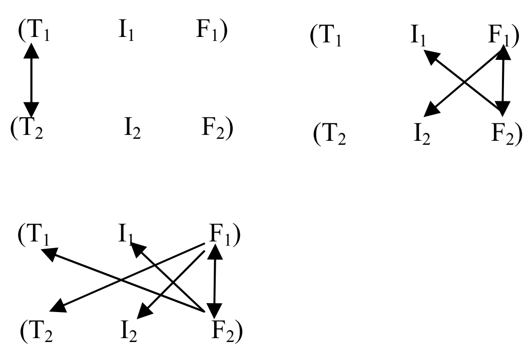
$$c_N^{TIF} : \left([0,1]\times[0,1]\times[0,1]\right)^2 \rightarrow [0,1]\times[0,1]\times[0,1]$$

$$c_N^{TIF}(x,y) = \left(T_1T_2, I_1I_2 + I_1T_2 + T_1I_2, F_1F_2 + F_1I_2 + F_1T_2 + F_2T_1 + F_2I_1\right)$$
.

The neutrosophic conjunction (intersection) operator $x \wedge_N y$ component truth, indeterminacy, and falsehood values result from the multiplication

$$\left(T_1 + I_1 + F_1\right)\cdot\left(T_2 + I_2 + F_2\right)$$

since we consider in a prudent way T p I p F , where "p " is a **neutrosophic relationship** and means "weaker", i.e. the products T_iI_j will go to I , T_iF_j will go to F , and I_iF_j will go to F for all i, j $\in$ {1,2}, i $\neq$ j, while of course the product T_1T_2 will go to T, I_1I_2 will go to I, and F_1F_2 will go to F (or reciprocally we can say that F prevails in front of I which prevails in front of T , and this neutrosophic relationship is transitive):



So, the truth value is T_1T_2 , the indeterminacy value is $I_1I_2 + I_1T_2 + T_1I_2$ and the false value is $F_1F_2 + F_1I_2 + F_1T_2 + F_2T_1 + F_2I_1$. The norm of x Ù$_N y$ is $(T_1 + I_1 + F_1)\times(T_2 + I_2 + F_2)$. Thus, if x and y are normalized, then x Ù$_N$ y is also normalized. Of course, the reader can redefine the neutrosophic conjunction operator, depending on application, in a different way, for example in a more optimistic way, i.e. I p T p F or T prevails with respect to I , then we get:

$$c_N^{ITF}(x,y) = \left(T_1T_2 + T_1I_2 + T_2I_1, I_1I_2, F_1F_2 + F_1I_2 + F_1T_2 + F_2T_1 + F_2I_1\right)$$

Or, the reader can consider the order T p F p I , etc.

V. ROBOT POSITION CONTROL BASED ON KINEMATICS EQUATIONS

A robot can be considered as a mathematical relation of actuated joints which ensures coordinate transformation from one axis to the other connected as a serial link manipulator where the links sequence exists. Considering the case of revolute-geometry robot all joints are rotational around the freedom ax [4, 5]. In general having a six degrees of freedom the manipulator mathematical analysis becomes very complicated. There are two dominant coordinate systems: Cartesian coordinates and joints coordinates. Joint coordinates represent angles between links and link extensions. They form the coordinates where the robot links are moving with direct control by the actuators.

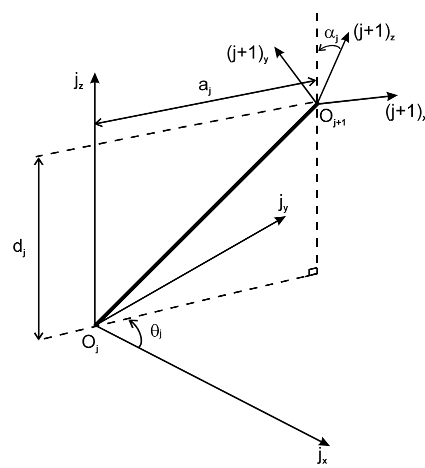


Fig.1. The robot control through DH transformation.

The position and orientation of each segment of the linkage structure can be described using Denavit-Hartenberg [DH] transformation [6]. To determine the D-H transformation matrix (Fig. 1) it is assumed that the *Z*-axis (which is the system's axis in relation to the motion surface) is the axis of rotation in each frame, with the following notations: θ_j - joint angled is the joint angle positive in the right hand sense about j_Z ; a_j - link length is the length of the common normal, positive in the direction of $(j+1)_X$; α_j - twist angled is the angle between j_Z and $(j+1)_Z$, positive in the right hand sense about the common normal ; d_j - offset distance is the value of j_Z at which the common normal intersects j_Z ; as well if j_X and $(j+1)_X$ are parallel and in the same direction, then $\theta_j = 0$; $(j+1)_X$ - is chosen to be collinear with the common normal between j_Z and $(j+1)_Z$ [7, 8] . Figure 1 illustrates a robot position control based on the Denavit-Hartenberg transformation. The robot joint angles, θ_c, are transformed in X_c - Cartesian coordinates with D-H transformation. Considering that a point in j, respectively j+1 is given by:

$$\begin{vmatrix} X \\ Y \\ Z \\ 1 \end{vmatrix}_j = {}^{j}P \quad \text{and} \quad \begin{vmatrix} X \\ Y \\ Z \\ 1 \end{vmatrix}_{j+1} = {}^{j+1}P' \tag{1}$$

then ${}^{j}P$ can be determined in relation to ${}^{j+1}P$ through the equation :

$${}^{j}P = {}^{j}A_{j+1} \cdot {}^{j+1}P, \tag{2}$$

where the transformation matrix ${}^{j}A_{j+1}$ is:

$${}^{j}A_{j+1} = \begin{bmatrix} \cos\theta_j & -\sin\theta_j \cdot \cos\alpha_j & +\sin\theta_j \cdot \sin\alpha_j & a_j \cdot \cos\alpha_j \\ \sin\theta_j & -\cos\theta_j \cdot \cos\alpha_j & -\cos\theta_j \cdot \sin\alpha_j & a_j \cdot \sin\alpha_j \\ 0 & \sin\theta_j & \cos\theta_j & d_j \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Control through forward kinematics consists of the transformation of robot coordinates at any given moment, resulting directly from the measurement transducers of each axis, to Cartesian coordinates and comparing to the desired target's Cartesian coordinates (reference point). The resulting error is the difference of position, represented in Cartesian coordinates, which requires changing. Using the inverted Jacobean matrix ensures the transformation into robot coordinates of the position error from Cartesian coordinates, which allows the generating of angle errors for the direct control of the actuator on each axis.

The control using forward kinematics consists of transforming the actual joint coordinates, resulting from transducers, to Cartesian coordinates and comparing them with the desired Cartesian coordinates. The resulted error is a required position change, which must be obtained on every axis. Using the Jacobean matrix inverting it will manage to transform the change in joint coordinates that will generate angle errors for the motor axis control.

Figure 2 illustrates a robot position control system based on the Denavit-Hartenberg transformation. The robot joint angles, θ_c, are transformed in X_c - Cartesian coordinates with D-H transformation, where a matrix results from (1) and (2) with θ_j -joint angle, d_j -offset distance, a_j - link length, α_j - twist.

Position and orientation of the end effector with respect to the base coordinate frame is given by X_C :

$$X_C = A_1 \cdot A_2 \cdot A_3 \cdot \ldots\ldots\ldots \cdot A_6 \tag{3}$$

Position error ΔX is obtained as a difference between desired and current position. There is difficulty in controlling robot trajectory, if the desired conditions are specified using position difference ΔX with continuously measurement of current position $\theta_{1,2,\ldots..6}$.

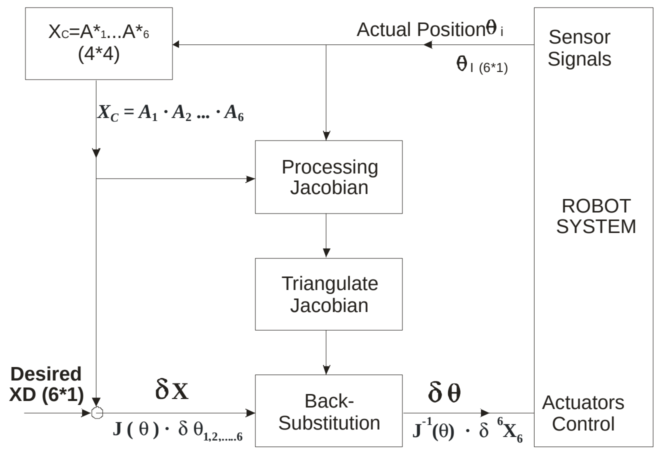


Fig. 2. Robot position control system based on the Denavit-Hartenberg transformation

The relation, between given by end-effector's position and orientation considered in Cartesian coordinates and the robot joint angles $\theta_{1,2,\ldots..6}$, it is :

$$x_i = f_i(\theta) \tag{4}$$

where θ is vector representing the degrees of freedom of robot. By differentiating we will have: $\delta\, {}^{6}X_6 = J(\theta) \cdot \delta\theta_{1,2,\ldots..6}$, where $\delta\, {}^{6}X_6$ represents differential linear and angular changes in the end effector at the currently values of X_6 and $\delta\theta_{1,2,\ldots..6}$ represents the differential change of the set of joint angles. $J(\theta)$ is the Jacobean matrix in which the elements a_{ij} satisfy the relation: $a_{ij} = \delta f_{i\text{-}1} / \delta\theta_{j\text{-}1}$, (x.6) where i, j are corresponding to the dimensions of x respectively θ. The inverse Jacobian transforms the Cartesian position $\delta^6 X_6$ respectively ΔX in joint angle error ($\Delta\theta$): $\delta\theta_{1,2,\ldots6} = J^{-1}(\theta) \cdot \delta\, {}^{6}X_6$.

VI. HYBRID POSITION AND FORCE CONTROL OF ROBOTS

Hybrid position and force control of industrial robots equipped with compliant joints must take into consideration the passive compliance of the system. The generalized area where a robot works can be defined in a constraint space with six degrees of freedom (DOF), with position constrains along the normal force of this area and force constrains along the tangents. On the basis of these two constrains there is described the general scheme of hybrid position and force control in figure 3. Variables X_C and F_C represent the Cartesian position and the Cartesian force exerted onto the environment. Considering X_C and F_C expressed in specific frame of coordinates, its can be determinate selection matrices S_x and S_f, which are diagonal matrices with 0 and 1 diagonal elements, and which satisfy relation: $S_x + S_f = I_d$, where S_x and S_f are methodically deduced from kinematics constrains imposed by the working environment [9, 10].

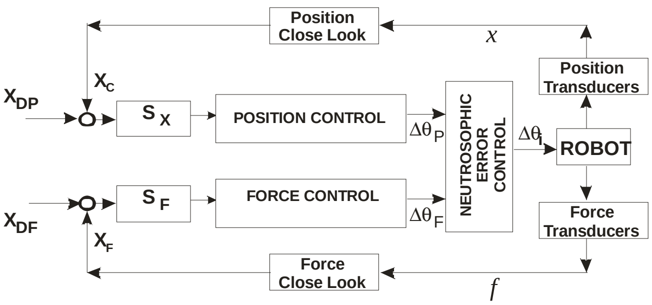


Fig. 3. General structure of hybrid control.

Mathematical equations for the hybrid position-force control. A system of hybrid position–force control normally achieves the simultaneous position–force control. In order to determine the control relations in this situation, ΔX_P – the measured deviation of Cartesian coordinate command system is split in two sets: ΔX^F corresponds to force controlled component and ΔX^P corresponds to position control with axis actuating in accordance with the selected matrixes S_f and S_x. If there is considered only positional control on the directions established by the selection matrix S_x there can be determined the desired end - effector differential motions that correspond to position control in the relation: $\Delta X_P = K_P \, \Delta X^P$, where K_P is the gain matrix, respectively desired motion joint on position controlled axis: $\Delta\theta_P = J^{-1}(\theta) \cdot \Delta X_P$ [11, 12].

Now taking into consideration the force control on the other directions left, the relation between the desired joint motion of end-effector and the force error ΔX_F is given by the relation: $\Delta\theta_F = J^{-1}(\theta) \cdot \Delta X_F$, where the position error due to force ΔX_F is the motion difference between ΔX^F– current position deviation measured by the control system that generates position deviation for force controlled axis and ΔX_D – position deviation because of desired residual force. Noting the given desired residual force as F_D and the physical rigidity K_W there is obtained the relation: $\Delta X_D = K_W^{-1} \cdot F_D$.

Thus, ΔX_F can be calculated from the relation: $\Delta X_F = K_F (\Delta X^F - \Delta X_D)$, where K_F is the dimensionless ratio of the stiffness matrix. Finally, the motion variation on the robot axis matched to the motion variation of the end-effectors is obtained through the relation: $\Delta\theta = J^{-1}(\theta) \, \Delta X_F + J^{-1}(\theta) \, \Delta X_P$. Starting from this representation the architecture of the hybrid position – force control system was developed with the corresponding coordinate transformations applicable to systems with open architecture and a distributed and decentralized structure.

For the fusion of information received from various sensors, information that can be conflicting in a certain degree, the robot uses the fuzzy and neutrosophic logic or set [3]. In a real time it is used a neutrosophic dynamic fusion, so an autonomous robot can take a decision at any moment.

CONCLUSION

In this paper we have provided in the first part an introduction to the neutrosophic logic and set operators and in the second part a short description of mathematical dynamics of a robot and then a way of applying neutrosophic science to robotics. Further study would be done in this direction in order to develop a **robot neutrosophic control**.

REFERENCES

[1] Florentin Smarandache, *A Unifying Field in Logics: Neutrosophic Field,* Multiple-Valued Logic / An International Journal, Vol. 8, No. 3, 385-438, June 2002

[2] Andrew Schumann, *Neutrosophic logics on Non-Archimedean Structures*, Critical Review, Creighton University, USA, Vol. III, 36-58, 2009.

[3] Xinde Li, Xianzhong Dai, Jean Dezert, Florentin Smarandache, *Fusion of Imprecise Qualitative Information*, Applied Intelligence, Springer, Vol. 33 (3), 340-351, 2010.

[4] Vladareanu L, *Real Time Control of Robots and Mechanisms by Open Architecture Systems*, cap.14, Advanced Engineering in Applied Mechanics, Eds-. T.Sireteanu,L.Vladareanu, Ed. Academiei 2006, pp.38, pg. 183-196, ISBN 978-973-27-1370-9

[5] Vladareanu L, Ion I, Velea LM, Mitroi D, Gal A,*"The Real Time Control of Modular Walking Robot Stability*", Proceedings of the 8th International Conference on Applications of Electrical Engineering (AEE '09), Houston, USA, 2009,pg.179-186, ISSN: 1790-5117 , ISBN: 978-960-474-072-7

[6] Denavit J., Hartenberg RB - *A kinematics notation for lower-pair mechanism based on matrices*. ASMEJ. Appl. Mechanics, vol.23June 1955, pg.215-221

[7] Sidhu G.S. – *Scheduling algorithm for multiprocessor robot arm control*, Proc. 19th Southeastern Symp., March,1997

[8] Vladareanu L; Tont G; Ion I; Vladareanu V; Mitroi D; *Modeling and Hybrid Position-Force Control of Walking Modular Robots*; American Conference on Applied Mathematics, pg:510-518; Harvard Univ, Cambridge, Boston, USA, 2010; ISBN: 978-960-474-150-2.

[9] L.D.Joly, C.Andriot, V.Hayward, *Mechanical Analogic in Hybrid Position/Force Control*, IEEE Albuquerque, New Mexico, pg. 835-840, April 1997

[10] Vladareanu L, *The robots' real time control through open architecture systems,* cap.11, Topics in Applied Mechanics, vol.3, Ed.Academiei 2006, pp.460-497, ISBN 973-27-1004-7

[11] Vladareanu L, Sandru OI, Velea LM, YU Hongnian, *The Actuators Control in Continuous Flux using the Winer Filters*, Proceedings of Romanian Academy, Series A, Volume: 10 Issue: 1 Pg.: 81-90, 2009, ISSN 1454-9069

[12] Yoshikawa T., Zheng X.Z. - *Coordinated Dynamic Hybrid Position/Force Control for Multiple Robot Manipulators Handling One Constrained Object*, The International Journal of Robotics Research, Vol. 12, No. 3, June 1993, pp. 219-230