ICANN '93

Proceedings of the International Conference
on Artificial Neural Networks
Amsterdam, The Netherlands
13–16 September 1993

Edited by
Stan Gielen and Bert Kappen

Springer-Verlag
London Berlin Heidelberg New York
Paris Tokyo Hong Kong
Barcelona Budapest

A Dynamical Model for the Generation of Curved Trajectories

P. Morasso†, V. Sanguineti†, and T. Tsuji‡

† *Department of Informatics, Systems, and Telecommunications, Genoa University, Italy*

‡ *Faculty of Engineering, Hiroshima University, Japan*

Abstract - In the framework of a central hypothesis of kinematic invariance, we propose a model (ξ-model) which is a non-linear dynamical system capable of generating, as motor primitives, a family of curved trajectories. The model links shape and speed by means of a suitable time base generator that drives two equations: a linear-speed equation and a turning-speed equation.

1. Introduction

The kinematics of human movements has remarkably invariant features in a wide range of timing, loading, and "postural" conditions. The question, then, is what we can infer about the underlying neural processes. The first dichotomy is between models (like the λ-model [3]) that privilege the properties of the musculo-skeletal apparatus and models that attribute the main morphogenetic role to more central planning processes. Although we think that the viscous-elastic properties of muscles are essential for skilled movements, particularly as regards fine compliance control, the *peripheral hypothesis* of kinematic invariances is not powerful enough, in our opinion, to capture the complexity of the topic and, in any case, it leaves open the question of where and how complex muscle patterns are generated.

For a *central hypothesis* of kinematic invariance, on the other hand, there is the opposite danger of reducing it to an abstract curve fitting exercise, quite uncoupled from the musculo-skeletal reality. From this point of view, we think that the central generator of kinematic patterns should be viewed as a *dynamical system* and not as some kind of *static* mechanism, like in the minimum-jerk model [4] or the power-law model [6]. Moreover, there is a *compositionality* problem: since complex trajectories are obviously composed as ordered sequences of discrete motor commands, which are the basic primitives? *Global* models, like the power law model, are good (perhaps, too good) for complex, endless trajectories but are not plausible for simple reaching movements, whereas *local* models, like the minimum-jerk model, (over-)privilege straight trajectories.

The model that we propose (ξ-model) is somehow in the middle: it is a non-linear dynamical system that, in the framework of a central hypothesis of kinematic invariance, is capable of generating, as motor primitives, a family of curved trajectories that include straight lines as special cases. The dynamics of the model quite constraints the range of possible dynamic behaviours: differently from the VITE model [2], that can generate any kind of speed profile by an appropriate choice of the $\gamma(t)$ function ($\dot{x}(t) = \gamma(t)(x_f - x)$), the ξ-model is intrinsically based on a symmetric mechanism and it only has 3 free parameters of immediate cognitive significance.

2. The mathematical model

The shape of a trajectory depends on the way in which curvature varies with the curvilinear coordinate. The model links shape and speed by means of a suitable time base generator that drives two equations: a linear-speed equation and a turning-speed equation.

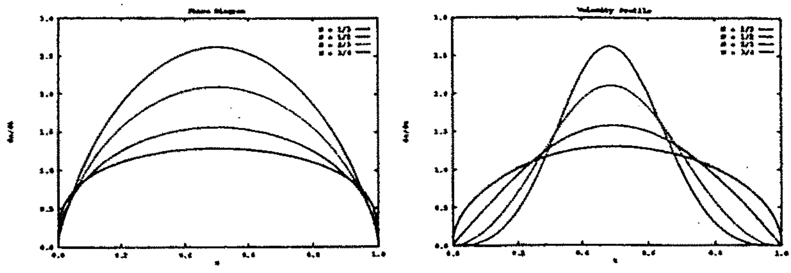

Figure 1: System dynamics corresponding to different values of e

2.1. The time base generator

The time base generator is a scalar dynamical system $\dot{\xi} = f(\xi)$ in the normalized variable ξ which is supposed to generate smooth sigmoidal signals $\xi = \xi(t)$ from $\xi(0) = 0$ to $\xi(T_f) = 1$ with a bell-shaped velocity profile and a desired (finite) movement duration T_f [1].

A class of models $f(\xi)$ that satisfy these conditions is given by $f(\xi) = \gamma\,[\xi(1-\xi)]^e$ with $e \in (0,1)$ [2]. Figure 1 plots the phase space diagram and the velocity profile corresponding to different values of e. As regards the exponent e, it can be shown that the condition $e \geq 2/3$ is necessary for the 3^{rd} time derivative of $\xi(t)$ (*jerk*) to be defined at $t = 0$ and $t = T_f$. In these conditions, the equilibrium configurations of the dynamical system do not satisfy the Lipschitz condition because $|df/d\xi| \rightarrow \infty$: $\xi = 1$ behaves as a *terminal attractor* and $\xi = 0$, which is unstable, is a *terminal repeller* [1]. In summary, the time base generator used in the simulation is described by the equation:

$$\dot{\xi} = \gamma[\xi(1-\xi)]^{2/3} \tag{1}$$

Remarkably, this model can be approximated with a simple neural network, presented in Figure 2, which consists of 4 neurons: $N1, N2, N3, N4$. $N1$ and $N2$ are additive neurons, $N3$ and $N4$ are multiplicative neurons. All of them have a sigmoidal activation function with high gain ($y = g(x) = (1-e^{-kx})/(1+e^{-kx})$, with $k >> 1$). $N3$ has a sufficiently long time constant for approximating it with an integrator, while the others have sufficiently short time constants to consider only the steady-state components. The network is a dynamical system in the ξ state variable and it is easy to derive the following state equation: $\dot{\xi}(t) = \gamma g(g(\xi)g(1-\xi))$. Since the sigmoidal function is monotonic, the state function $g(g(\xi)g(1-\xi))$ can be approximated by a function $f(\xi(1-\xi))$ that, similarly to a power function with an exponent smaller than unity, is very steep in the origin and thus guarantees a finite T_f. The γ parameter has a double function: it controls the speed and it can be used to reset the time base generator and make it excitable for subsequent activation cycles.

2.2. Generation of curved trajectories

The linear speed v and the turning speed ω can be linked to a common goal (reaching a target with a desired orientation) by using the same time base generator and a pair of error measures: the linear error Δr and the angular error $\Delta\theta$. The former one is simply the norm of the difference between the final and the current position: $\|x_f - x(t)\|$. With this notation, it is possible to demonstrate that the following linear speed equation

$$v = \Delta r\, \dot{\xi}/(1-\xi) \tag{2}$$

[1] $\dot{\xi}$ should tend to zero for $t \rightarrow 0$ and $t \rightarrow T_f$. T_f is given by $\int_0^1 d\xi/f(\xi)$ and a sufficient condition for attaining a finite value is that $f(\xi)$ is infinitesimal of order n, $n < 1$, for both $\xi \rightarrow 0$ and $\xi \rightarrow 1$.

[2] For this class of functions, it can be shown that movement duration T_f is inversely proportional to the gain factor γ: $\gamma(e) = 1/T_f \Gamma^2(1-e)/\Gamma(2-2e)$

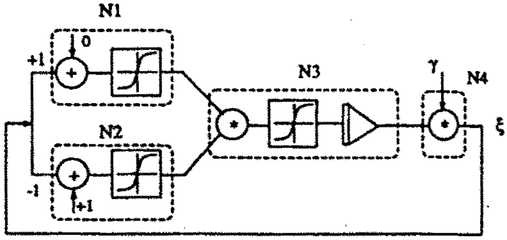

Figure 2: Neural time base generator

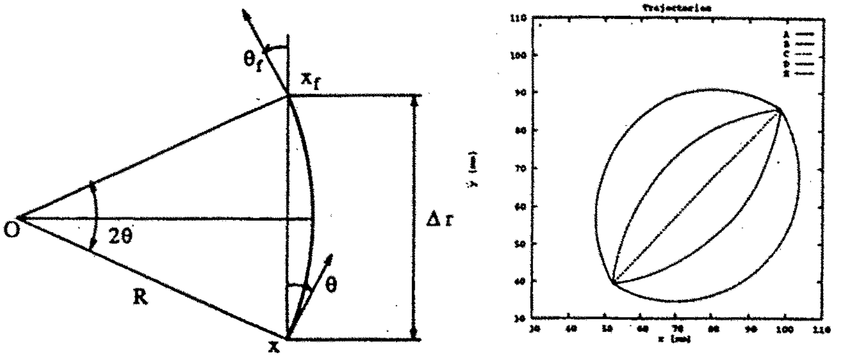

Figure 3: Circular trajectory (left) and simulated trajectories (right)

allows $x(t)$ to reach x_f at the same time in which $\xi(t)$ reaches 1, provided that the turning speed equation allows Δr to decrease in a monotonic way. In the limit case of a straight trajectory, this is equivalent to a simple interpolation with no rotation: $x(t) = x_0 + (x_f - x_0)\xi(t)$ [3]. For general movements, in which the initial and/or the final directions are not aligned with $x_f - x_0$ vector, a sufficient condition for the trajectory to hit the target with the right direction is that at some point in the trajectory (i.e. for $\xi < 1$), the running point $x = x(t)$ reaches the configuration which characterizes a circular motion: $\theta(t) = -\theta_f$ (see Fig. 3, left). In this case, the following relation between ω and v holds: $\omega = v/R$ (R is the radius of the circle) and since $\Delta r = 2R\sin\theta$, we get $\omega = 2\sin\theta\,\dot{\xi}/(1-\xi)$. In general, we can write a turning speed equation

$$\omega = \Delta\theta\dot{\xi}/(1-\xi) \tag{3}$$

where the angular error term is given by $\Delta\theta(t) = 2\sin\theta(t)$ in the limit case of a circular motion. If the current conditions do not support a circular motion, then a possible strategy is to smoothly drive $\theta(t)$ towards such symmetric condition. The two criteria (approaching the condition of symmetry and following a circular path) can be combined in the following hierarchical way ($\Delta\theta_{sym} = -(\theta + \theta_T)$):

$$\Delta\theta = \Delta\theta_{sym} + 2\sin\theta e^{-\Delta\theta_{sym}^2/2\sigma^2} \tag{4}$$

It can be seen that while the error term for symmetry is large, the equation reduces to $\Delta\theta = \Delta\theta_{sym}$. This equation works just like a feedback control law to attain a symmetric configuration where the target and present directions of the trajectory are symmetric with respect to the line between the initial and target points. Once the symmetric configuration is attained, the symmetry error vanishes and the motion equation of the angular velocity ω reduces to $\Delta\theta = 2\sin\theta$. Consequently,

[3]It is interesting to note that, in this form, if we put $\gamma(t) = \dot{\xi}/(1-\xi)$, the ξ-model is a special case of VITE.

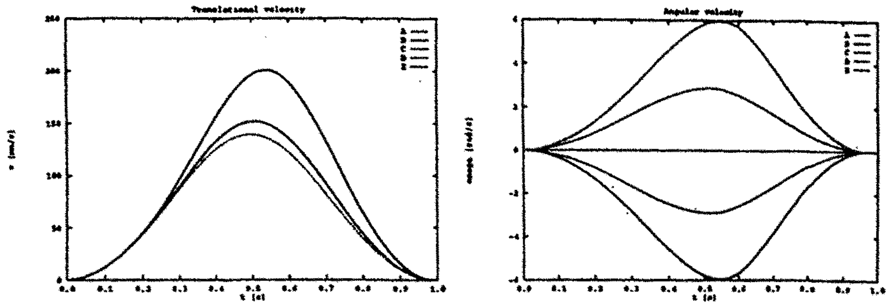

Figure 4: Simulated translational velocity (left) and angular velocity (right)

once the symmetric configuration is attained, the ξ-model can generate the circular trajectory. The trajectory terminates at the target point where the translational velocity becomes zero. This guarantees the convergency of the ξ-model for curved movements. The value of σ determines the relative weights of the two components of the error function. To verify the convergency of the ξ-model, computer simulations have been performed. Fig. 3, right shows examples of the generated trajectories by the ξ-model where the 2/3 power law is used. The initial and target points are fixed, while several initial and target directions satisfying the symmetric configuration condition are used. The corresponding translational and angular velocities of the generated trajectories are indicated in Fig. 4. The spatial profiles of the generated trajectories are almost circular and the velocity profiles are almost bell-shaped. The ξ-model can generate the curved trajectory as well as the straight trajectory as a single stroke movement.

3. Conclusions

In conclusion, the ξ-model consists of three equations: the time base generator (1) and the two speed equations (2,3). It can generate a large class of curved movements, parametrized by Δr_0, θ_0 and θ_f. In a preliminary experiments, in which we recorded hand movements with a digitizing tablet, we estimated the range of values of the angular differences that produce trajectories with unimodal speed profiles. We are also in the course of integrating the ξ-model, which dictates the behaviour of the *end effector*, with a self-organized neural model [5] of a kinematic chain which is actually carrying out the multi-joint coordination task.

References

[1] J. Barhen, S. Gulati, and M. Zak. Neural learning of constrained nonlinear transformations. *IEEE Computer*, 6:67–76, 1989.

[2] D. Bullock and S. Grossberg. VITE and FLETE: Neural modules for trajectory formation and postural control. In W.A. Hershberger, editor, *Volitional Action*, pages 253–297. North-Holland/Elsevier, Amsterdam, 1989.

[3] A.G. Feldman. Functional tuning of the nervous system with control of movement or maintenance of a steady posture. ii. controllable parameters of the muscle. *Biophysics*, 11:565–578, 1966.

[4] T. Flash and N. Hogan. The coordination of arm movements: an experimentally confirmed mathematical model. *Neuroscience*, 7:1688–1703, 1985.

[5] P. Morasso and V. Sanguineti. SOBoS – A Self-Organizing Body Schema. In I. Aleksander and J. Taylor, editors, *Artificial Neural Networks*, Amsterdam, 1992. North-Holland.

[6] P. Viviani and C. Terzuolo. Trajectory determines movement dynamics. *Neuroscience*, 7:431–437, 1982.