# Discretely Actuated Manipulator Workspace Generation by Closed Form Convolution[1]

**Imme Ebert-Uphoff**[2]
Graduate Student.

**Gregory S. Chirikjian**
Associate Professor.

Department of Mechanical Engineering
Johns Hopkins University
Baltimore, MD 21218

*We determine workspaces of discretely actuated manipulators using convolution of real-valued functions on the Special Euclidean Group. Each workspace is described in terms of a density function that provides the number of reachable frames inside any unit volume of the workspace. A manipulator consisting of n discrete actuators each with K states can reach $K^n$ frames in space. Given this exponential growth, brute force representation of discrete manipulator workspaces is not feasible in the highly actuated case. However, if the manipulator is of macroscopically-serial architecture, the workspace can be generated by the following procedure: (1) partition the manipulator into B kinematically independent segments; (2) approximate the workspace of each segment as a continuous density function on a compact subset of the Special Euclidean Group; (3) approximate the whole workspace as a B-fold convolution of these densities. We represent density functions as finite Hermite-Fourier Series and show for the planar case how the B-fold convolution can be performed in closed form. If all segments are identical only O(log B) convolutions are necessary.*

## 1 Introduction

The concept of a convolution product of real-valued functions on the Special Euclidean Group (which describes rigid body motion in Euclidean space) is examined in this paper. The primary application discussed here is the generation of discretely actuated manipulator workspaces in terms of the *density* of reachable frames. Density is a function that describes the number of reachable frames inside any volume[2] in the workspace. Figure 1 shows a discrete manipulator in three different configurations superposed on a discretized representation of the workspace density. Dark gray corresponds to high density, i.e., a large number of reachable positions per unit area, and the boundary between gray and white marks the border of the workspace. In the context of discrete actuation, the density function in many ways replaces dexterity measures as a scalar function of importance defined over the workspace. The reason is that the density in the neighborhood of a given point/frame is an indication of how accurately a discretely actuated manipulator can reach that point/frame. Hence, important applications of workspace generation in the context of discrete manipulators are (a) the design of manipulators with prescribed workspace density (e.g., to prescribe areas of high accuracy as described in Chirikjian (1996)) (b) the use of workspace densities for inverse kinematics, as described in Ebert-Uphoff and Chirikjian (1996a).

To compute this workspace density function using brute force, i.e., calculating all possible configurations, is computationally intractable. For a manipulator with $n$ actuators each with $K$ states brute force calculation requires $\mathbf{O}(K^n)$ computations.

To give a numerical example we consider the variable geometry truss manipulator shown in Fig. 1. It consists of ten variable geometry truss modules, each of which has three actuated links. Each actuated link is assumed to have only two stable states. Fig. 2 shows a schematic of all possible configurations of one truss module. The whole manipulator in Fig. 1 has a total of $2^{(3\cdot 10)} = 2^{30} \approx 10^9$ states. Figure 3 shows a prototype consisting of five such truss modules that was built in our laboratory. In Section 4 we generate the workspace density of a manipulator consisting of eight truss modules of the same type but each actuator is assumed to have four different states instead of two, i.e., the whole manipulator has a total of $4^{(3\cdot 8)} = 4^{24} \approx 2.8 \times 10^{14}$ states. In our estimation it would require more than a decade on an average workstation to compute all of these configurations using brute force.

In this paper we propose an alternative way to approximate the workspace densities for manipulators of macroscopically-serial architecture. A macroscopically-serial manipulator can be described as a serial collection of modules, as shown in Fig. 4, where each module is mounted on top of the previous one. The modules can be serial links or parallel structures, such as Gough (Stewart) platforms. For this kind of manipulator the workspace density can be generated by the following procedure: (1) partition the manipulator into $B$ kinematically-independent segments; (2) approximate the workspace of each segment as a continuous density function on a compact subset of the Special Euclidean Group; (3) approximate the whole workspace as a repeated $B$-fold convolution of these densities. In this article we represent density functions as finite Hermite-Fourier Series and show for the planar case how the $B$-fold convolution can then be performed in closed form based only on the coefficients of the series. If all segments are identical only $\mathbf{O}(\log B)$ convolutions are necessary.

In the following section, we review work that deals with manipulator workspace generation. Section 3 reviews how the convolution product of functions on $SE(D)$ applies to workspace generation and presents an implementation of the convolution in closed form. Section 4 presents numerical results for workspace generation using the concept of convolution. Section 5 presents a final discussion and conclusions.

## 2 Literature Review

The study of workspace generation started with serial open kinematic chains. See for example Kumar and Waldron

[1] This work was sponsored by National Science Foundation grant #IRI-9357738; and a 1994 Presidential Faculty Fellows Award.

[2] Currently a Postdoctoral Researcher at Laval University, Quebec City, P.Q., Canada.

[3] The workspace for positioning and orienting an object in three dimensional space is six dimensional.

Contributed by the Mechanisms Committee for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received April 1996. Associate Technical Editor: Eugene Fichter.

  

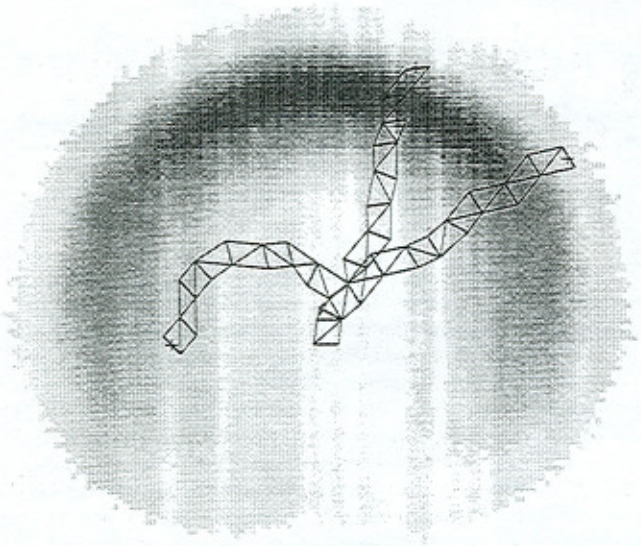

Fig. 1 A discretely actuated manipulator with $2^{30}$ states

(1981b), Gupta and Roth (1982), Yang and Lee (1983) and Tsai and Soni (1981). These works contributed to a framework for workspace generation and derived analytic descriptions for the workspace of certain manipulators. Other works include Lai and Menq (1988) and Emiris (1993). Works focusing on manipulators with only revolute joints include Ceccarelli and Vincigyerra (1995), Korein (1985), Kwon et al. (1994) and Selfridge (1983).

A number of different approaches exist for manipulators of more general architecture. Alciatore and Ng (1994) propose an approach that generates workspaces based on the Monte-Carlo method. Rastegar and Deravi (1987) divide the workspace of a manipulator into subsets according to the number of solutions of the inverse kinematics problem which are calculated numerically using the Jacobian.

A number of works have dealt with the optimal design of workspaces. E.g., Dwarakanath et al. (1994) design a manipulator to contain a desired rectangle in the workspace, while Gosselin and Guillot (1991) optimize the workspace to have a boundary specified by a collection of arcs.

Blackmore and Leu (1992) discuss the swept volume for an object (a set in $\mathbb{R}^n$) under rigid body motion. The method is applied to generate the workspace of serial open kinematic chains. Park and Brocket (1994) derive dexterity measures for mechanisms over the workspace volume based on differential geometry and the theory of Lie groups.

Discrete manipulators were considered by Roth et al. (1973), Pieper (1968), and Koliskor (1986). Sen and Mruthyunjaya (1994) consider manipulators with only discrete joint states to model continuous-range-of-motion manipulators with limited joint resolution. To the best of our knowledge that work was

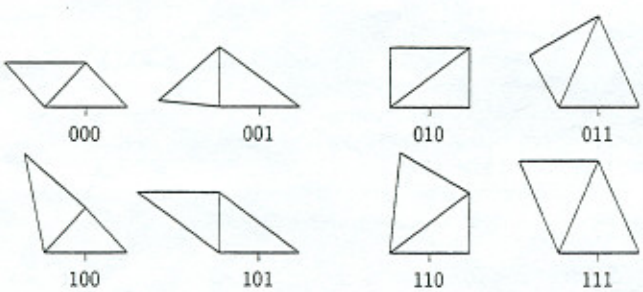


Fig. 2 The eight possible configurations of a truss with three bistable actuators

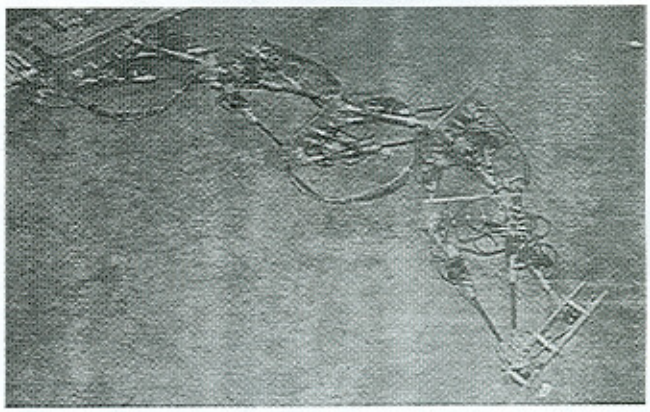

Fig. 3 Manipulator consisting of 5 truss modules

the first to point out the importance of the workspace density as a measure of local accuracy of a discretely actuated manipulator. Kumar and Waldron (1981a) describe a model that relates the joint accuracy of a continuously actuated manipulator to the accuracy of the end effector.

The density of points in the workspace can be used constructively for inverse kinematics of discrete manipulators (Ebert-Uphoff and Chirikjian, 1996a). Previously, we presented a method to efficiently generate workspaces (Ebert-Uphoff and Chirikjian, 1995) in terms of densities. While those works are based on a discrete representation of workspace densities similar to a histogram, we now take a continuous approach. The advantage is that we capture the density distribution of the whole workspace in a small number of coefficients, hence reducing the amount of data to be handled. Furthermore, the coefficients provide access to global properties of the workspace, e.g., center of mass, average orientation, etc.

In order to apply the method presented in this paper for hybrid-serial-parallel manipulators it is necessary to generate the workspace of parallel platform manipulators. A great deal of work on this topic was reported for example by Merlet (1992, 1995) and references therein. Kumar (1992) discusses the workspace for mechanisms consisting of serial chains mounted in parallel.

## 3 Workspace Generation by Convolution

We begin this section by reviewing how the workspace of a discretely actuated manipulator of macroscopically-serial structure can be generated using repeated convolution. This formulation of workspace generation motivates our study of closed-form calculation of the convolution product of functions on $SE(D)$ (the rigid motion group of $D$-dimensional Euclidean space), which is discussed in the remainder of this article.

A given discretely actuated manipulator is divided into $B$ kinematically-independent modules, each containing a number of actuators. As outlined in Section 1, modules can have a parallel kinematic structure internally, but the modules are all cascaded in a serial way. The modules are numbered from 1 to $B$, as shown in Fig. 4, starting at the base with module 1 and increasing up to the most distal module, module $B$. For each module one frame is attached to the base of the module and a second one to the top, where the next module is attached.

Since the manipulator is actuated discretely, each of these modules has only a finite number of states. Each state corresponds to a transformation $H \in SE(D)$ of the lower frame to the upper frame. The set of all possible end-effector frames can be generated by combining all possible frame transformations for all modules, but, as we already know, this is not feasible if the manipulator consists of many actuators or if each actuator has a large number of states.

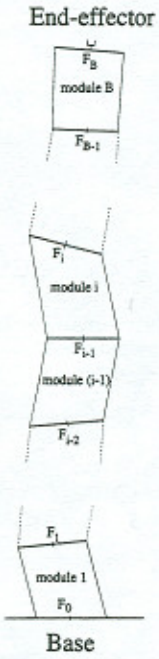


**Fig. 4 A macroscopically-serial manipulator**

Instead the frame distribution for each module is represented separately by dividing $SE(D)$ into small volume elements, counting the number of frames that lie in each of these volume elements, and dividing this number by the volume of each element. The result is a discrete density function (or histogram) for each of the $B$ modules: $\rho_1(H), \ldots, \rho_B(H)$. The concatenation of all possible frame transformations corresponds to the convolution of the density functions of all modules, i.e., the density function $\rho_W$ of the total workspace of the manipulator is obtained as

$$\rho_W(H) = (\rho_1 * \rho_2 * \ldots * \rho_B)(H).$$

If all modules of the manipulator are identical then $\rho_2 = \rho_1 * \rho_1$ is the workspace density of a two-module segment, $\rho_4 = \rho_2 * \rho_2$ is the density of a four-module segment, $\rho_8 = \rho_4 * \rho_4$ is the density of an eight-module segment, etc. This means that the workspace density of a manipulator consisting of $B = 2^m$ modules can be calculated using $m = \log_2 B$ convolutions.

### 3.1 Convolution of Functions on SE(2)

*3.1.1 Definition of Convolution on SE(D).* The concept of convolution can be formulated in a general way for functions on Lie groups. This general form is often used in mathematical physics (see Vilenkin and Klimyk (1991)).

The set of rigid body motions, $SE(D)$, with matrix multiplication as group operation, is a Lie group. For any two functions $\rho_1(H), \rho_2(H) \in \mathcal{L}^2(SE(D))$,[3] the convolution product is defined as

$$(\rho_1 * \rho_2)(H) = \int_{SE(D)} \rho_1(\mathcal{H})\rho_2(\mathcal{H}^{-1}H)d\mu(\mathcal{H}),$$

where $d\mu(H)$ is a left and right invariant volume element (see Park and Brockett (1994) and Murray et al. (1994)). If we are dealing with homogeneous transformations in space, and use z-x-z-Euler angles to parametrize the rotation part of elements, $H \in SE(3)$, the volume element takes the form

$$d\mu(H(x_1, x_2, x_3, \phi, \theta, \psi)) = \sin\theta dx_1 dx_2 dx_3 d\phi d\theta d\psi.$$

In the planar case, using $x_1, x_2, \theta$ as parameters, we get

$$d\mu(H(x_1, x_2, \theta)) = dx_1 dx_2 d\theta.$$

Applying this parametrization for $SE(2)$ the homogeneous transforms $H$ and $\mathcal{H}$ take the forms:

$$H(x_1, x_2, \theta) = \begin{pmatrix} \cos\theta & -\sin\theta & x_1 \\ \sin\theta & \cos\theta & x_2 \\ 0 & 0 & 1 \end{pmatrix}$$

and

$$\mathcal{H}(\xi_1, \xi_2, \alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha & \xi_1 \\ \sin\alpha & \cos\alpha & \xi_2 \\ 0 & 0 & 1 \end{pmatrix}.$$

In the following we write $\rho(x_1, x_2, \theta)$ to denote the function $\rho(H)$ parametrized by $x_1, x_2$ and $\theta$, i.e., $\rho(x_1, x_2, \theta) = \rho(H(x_1, x_2, \theta))$. For this parametrization the convolution product on $SE(2)$ can be expressed in the form:

$$\begin{aligned}(\rho_1 * \rho_2)(x_1, x_2, \theta) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\pi}^{\pi} \rho_1(\xi_1, \xi_2, \alpha)\rho_2((x_1 - \xi_1)\cos\alpha \\ &\quad + (x_2 - \xi_2)\sin\alpha, -(x_1 - \xi_1)\sin\alpha \\ &\quad + (x_2 - \xi_2)\cos\alpha, \theta - \alpha)\, d\xi_1 d\xi_2 d\alpha. \quad (1)\end{aligned}$$

*3.1.2 Connection between Convolution and Workspace Generation.* To understand the connection between convolution of functions on $SE(D)$ and workspace generation let us consider a manipulator that consists of two kinematically-independent segments stacked on top of one another. The workspace density functions of the lower and upper segments are given as $\rho_1(\mathcal{H})$ and $\rho_2(H)$, respectively.

Each frame transformation can be expressed in terms of a homogeneous transformation matrix, where

- $\mathcal{H}$ denotes the transformations from base to top of the lower segment
- $H$ denotes the transformations from base to top of the upper segment and
- $H'$ denotes the transformations from base to top of the whole manipulator, i.e., $H' = \mathcal{H} \cdot H$.

The workspace density functions, $\rho_1$ and $\rho_2$, can be viewed as a measure of the frequency of occurrence of homogeneous transformations for the lower and upper segments, respectively. To calculate the frequency of occurrence $\rho_3(H')$ of a particular homogeneous transformation for the whole manipulator we have to:

- consider any configuration $\mathcal{H}$ that the lower segment can attain and weigh it by the frequency of occurrence for this segment, $\rho_1(\mathcal{H})$,
- calculate the configuration $H$ of the upper segment that yields $H'$ as the transformation for the whole manipulator, i.e., $H = \mathcal{H}^{-1} \cdot H'$, and weigh it by the frequency of occurrence for the upper segment, $\rho_2(\mathcal{H}^{-1} \cdot H')$,
- multiply these two terms to get the combined frequency of occurrence and add up all these contributions by integrating over all $\mathcal{H} \in SE(D)$:

$$\rho_3(H') = \int_{SE(D)} \rho_1(\mathcal{H})\rho_2(\mathcal{H}^{-1}H')d\mu(\mathcal{H}).$$

**Result:** This shows that the workspace density $\rho_3$ of a manipulator consisting of two segments stacked on top of each other is

[3] We denote $\mathcal{L}^2(SE(D))$ to be the space of all real-valued square-integrable functions on $SE(D)$, i.e., $\rho_i(H) \in \mathbb{R}$ for all $H \in SE(D)$, and $\int_{SE(D)} \rho_i^2(H)d\mu(H) < \infty$, where $d\mu(H)$ is an appropriately defined volume element.

just the convolution of the workspace density functions $\rho_1$ and $\rho_2$ of the segments. This operation is simply repeated to generate the density for the whole manipulator.

### 3.2 Implementation of Closed-Form Convolution

*3.2.1 Representation of Density Functions.* In this section we are looking for an efficient way to approximate a function $\rho(H) = \rho(x, y, \theta)$ with compact support in closed form. We consider an expansion of the form

$$\rho(H) = \sum_{i=0}^{N} \sum_{j=0}^{N} \sum_{k=0}^{N} c_{ijk} f_i(x) g_j(y) p_k(\theta). \tag{2}$$

To guarantee a good approximation of functions in $\mathcal{L}^2(SE(2))$ it is necessary that

- the set of functions $\{f_i(\cdot)\}$ and $\{g_i(\cdot)\}$ are complete in the set of square integrable functions on the real line $\mathbb{R}$,
- $\{p_i(\cdot)\}$ is complete in the set of bounded functions on the circle.

Further desired properties are:

- the convolution product should be form-preserving, i.e., the convolution of two series of the form of Eq. (2) should result in a series of the same form. (Without this condition, multiple convolutions would lead to expressions which could not be evaluated.)
- the functions $f_i(\cdot)$ and $g_i(\cdot)$ should decay rapidly to zero for large arguments ($|x| \gg 0$).
- each set of functions should be orthogonal, so that the constants $c_{ijk}$ can be determined independently. This also allows one to choose the level of detail by the number of terms retained.
- it should be possible to evaluate each function quickly to guarantee efficient evaluation of the series approximation.

**Our Choice:**
The following introduces a set of approximation functions which satisfies all the requirements. The remainder of this article is based on this choice. The functions $f_i(\cdot)$ and $g_i(\cdot)$ are chosen to be Hermite functions and the angular part of the distribution is represented in terms of trigonometric functions:

$$f_i(x) = h_i(x)$$

$$g_j(y) = h_j(y)$$

$$p_k(\theta) = \cos(k\theta + \psi_k) = a_k \cos(k\theta) + b_k \sin(k\theta)$$

In summary functions on $SE(2)$ are approximated as:

$$\rho(x, y, \theta) = \sum_{i,j,k=0,0,0}^{N,N,N} h_i(x) h_j(y) [a_{ijk} \cos k\theta + b_{ijk} \sin k\theta].$$

*3.2.2 Review of the Properties of Hermite Functions.* Normalized *Hermite functions* are derived from Hermite polynomials as (Vilenkin and Klimyk, 1991):

$$h_i(x) = \frac{H_i(x) e^{-(x^2/2)}}{(i! 2^i \sqrt{\pi})^{1/2}},$$

where the *Hermite polynomials* $H_i(x)$ are defined as

$$H_n(x) = (-1)^n e^{x^2} \frac{d^n}{dx^n} (e^{-x^2}).$$

In accordance with this definition Hermite polynomials can be expressed explicitly as

$$H_n(x) = \sum_{r=0}^{n} \epsilon_{nr} x^r,$$

$$\epsilon_{nr} = \begin{cases} \dfrac{(-1)^{(n-r)/2} n! 2^r}{\left(\dfrac{n-r}{2}\right)! r!} & \text{if } (r+n) \text{ even}, \\ 0 & \text{if } (r+n) \text{ odd}. \end{cases} \tag{3}$$

Figure 5 shows normalized Hermite functions $h_i(x)$ for several values of $i$. Hermite functions are suitable for approximating density functions because (1) they are complete in the space of all square integrable functions on the real line, (2) they rapidly decay to zero for $|x| \gg 0$, and (3) they are orthonormal with respect to the scalar product

$$(f \cdot g) = \int_{-\infty}^{\infty} f(x) g(x) dx.$$

A square integrable function $\alpha(x)$ can be approximated as a linear combination of Hermite functions as

$$\alpha(x) = \lim_{N \to \infty} \sum_{i=0}^{N} a_i h_i(x)$$

where, using the orthonormality of the Hermite functions, the coefficients $a_i$ are computed independently as

$$a_i = \int_{-\infty}^{\infty} \alpha(x) h_i(x) dx.$$

The resulting approximation for a finite number of terms, $N$, provides a least-squared approximation of the function.

To get a good approximation using a finite number of coefficients it is necessary to scale the range of $x$- and $y$- coordinates. The reason is that the density functions to be represented, as well as the Hermite functions used to represent them, are only significantly different from zero in a very restricted range of $x$- and $y$-values. These areas have to coincide as much as possible to allow a good approximation using only few terms of the Hermite-Fourier-Series. As explained in Ebert-Uphoff (1997) this can be achieved by

(1) scaling the input functions $\rho_1$ and $\rho_2$ using a common scale factor $L$,

$$\tilde{\rho}_1(x, y, \theta) = \rho_1(Lx, Ly, \theta), \quad \tilde{\rho}_2(x, y, \theta) = \rho_2(Lx, Ly, \theta),$$

(2) convolving the scaled functions $\tilde{\rho}_1$ and $\tilde{\rho}_2$ to get $\tilde{\rho}_3 = \tilde{\rho}_1 * \tilde{\rho}_2$, and
(3) calculating the desired function $\rho_3 = \rho_1 * \rho_2$ from $\tilde{\rho}_3$ using the relationship

$$\rho_3(x, y, \theta) = L^2 \tilde{\rho}_3\left(\frac{x}{L}, \frac{y}{L}, \theta\right).$$

Using this procedure we can take care of the scaling before the convolution, so that the scale factor $L$ does not have to be considered in the following calculations.

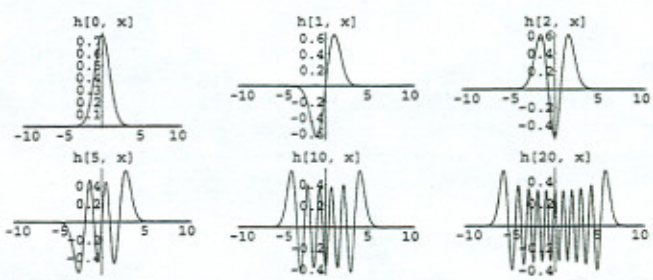


**Fig. 5 Normalized hermite functions $h_i(x)$ for $i$ = 0, 1, 2, 5, 10, 20**

*3.2.3 Fitting Procedure.* Given a discrete density histogram $\rho_{ijk}$ formed by dividing a compact subset of $SE(2)$ into blocks of equal size, and counting how many discrete frames lie in each block, a continuous density function is fit to this discrete data in the least squared sense by finding the constants $a_{ijk}$, $b_{ijk}$. Due to the orthonormality of Hermite functions, the coefficients are explicitly calculated as follows:

$$a_{ij0} = \frac{1}{2\pi}\int_{-\pi}^{\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\rho(x, y, \theta)h_i(x)h_j(y)dxdyd\theta,$$

$$b_{ij0} = 0,$$

$$a_{ijk} = \frac{1}{\pi}\int_{-\pi}^{\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\rho(x, y, \theta)h_i(x)h_j(y)\cos k\theta dxdyd\theta,$$

$$b_{ijk} = \frac{1}{\pi}\int_{-\pi}^{\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\rho(x, y, \theta)h_i(x)h_j(y)\sin k\theta dxdyd\theta.$$

If the data is provided in terms of a discrete density array that represents the number of points per pixel divided by pixel volume, the integration is approximated by a summation of the form:

$$a_{lmn} \approx \frac{1}{\pi}\sum_{i=0}^{2i_0}\sum_{j=0}^{2j_0}\sum_{k=0}^{k_0}\rho(x_i, y_j, \theta_k)h_l(x_i)h_m(y_j)\cos n\theta_k\Delta x\Delta y\Delta\theta$$

$$= \frac{\Delta x\Delta y\Delta\theta}{\pi}\sum_{i=0}^{2i_0}h_l(x_i)\sum_{j=0}^{2j_0}h_m(y_j)\sum_{k=0}^{k_0}\rho_{ijk}\cos n\theta_k$$

Hermite polynomials can be converted to power series of the same order (and vice versa) at any time without any loss of information according to Eq. (3). Substitution results in the following series representation for $\rho(x, y, \theta)$:

$$\rho(x, y, \theta) = e^{-(x^2+y^2)/2}\sum_{l,m=0}^{N}x^l y^m\sum_{k=0}^{N}(a'_{lmk}\cos k\theta + b'_{lmk}\sin k\theta),$$

where

$$a'_{lmk} = \sum_{i=l}^{N}\sum_{j=m}^{N}\frac{\epsilon_{il}\epsilon_{jm}}{(i!2^i j!2^j\pi)^{1/2}}a_{ijk},$$

$$b'_{lmk} = \sum_{i=l}^{N}\sum_{j=m}^{N}\frac{\epsilon_{il}\epsilon_{jm}}{(i!2^i j!2^j\pi)^{1/2}}b_{ijk}.$$

*3.2.4 Closed-Form Convolution of Hermite-Fourier Series.* Given two density functions on $SE(2)$ which are both represented using a series approximations of the form

$$\rho_1(x, y, \theta) = e^{-(x^2+y^2)/2^\nu}\sum_{i,j,k=0}^{N}x^i y^j(a'_{ijk}\cos k\theta + b'_{ijk}\sin k\theta),$$

$$\rho_2(x, y, \theta) = e^{-(x^2+y^2)/2^\nu}\sum_{l,m,n=0}^{N}x^l y^m(c'_{lmn}\cos n\theta + d'_{lmn}\sin n\theta),$$

their convolution as defined in Eq. (1) is evaluated as

$$(\rho_1*\rho_2)(x, y, \theta) = \int_{-\pi}^{\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}e^{-(\xi^2+\eta^2)/2^\nu}$$

$$\times e^{-((x-\xi)^2+(y-\eta)^2)/2^\nu}\sum_{i,j,k=0}^{N}\sum_{l,m,n=0}^{N}\xi^i\eta^j[(x-\xi)\cos\alpha$$

$$+ (y-\eta)\sin\alpha]^l[-(x-\xi)\sin\alpha + (y-\eta)\cos\alpha]^m$$

$$\times [a'_{ijk}\cos k\alpha + b'_{ijk}\sin k\alpha][c'_{lmn}\cos n(\theta-\alpha)$$

$$+ d'_{lmn}\sin n(\theta-\alpha)]d\xi d\eta d\alpha.$$

The integration in the convolution can be reduced to the integration of certain constant terms which all have a closed-form solution. These constants can be calculated off-line.

Writing the convolution product $(\rho_1*\rho_2)$ as a new series with coefficients $e'_{ijk}$, $f'_{ijk}$,

$$(\rho_1*\rho_2)(x, y, \theta) = e^{-(x^2+y^2)/2^{\nu+1}}\sum_{\lambda=0}^{3N}\sum_{\mu=0}^{3N}\sum_{n=0}^{N}x^\lambda y^\mu[e'_{\lambda\mu n}\cos n\theta + f'_{\lambda\mu n}\sin n\theta],$$

the new coefficients can be calculated from the input coefficients, ($a'_{ijk}$, $b'_{ijk}$, $c'_{ijk}$ and $d'_{ijk}$), based on the constants calculated off-line through simple algebraic operations. The derivation of this procedure extends over ten pages and can be found in Ebert-Uphoff (1997) and Ebert-Uphoff and Chirikjian (1996b).

The complexity of this procedure is of the order $\mathbf{O}(N^{10})$. While this may be a large computation, it is independent of the number of actuator states. Note, that the number of terms retained has increased to $3N \times 3N \times N$. In order to cut the number of terms to the original number of $N \times N \times N$ terms the series has to be transformed back to Hermite-polynomials. The complexity of this process is negligible as compared to the complexity of the convolution.

## 4 Numerical Results

In this section we present a numerical example of workspace generation using the method presented in this paper. We consider the following task: given the workspace density of a planar manipulator that consists of four truss modules each with 64 states, calculate the workspace density of a planar manipulator of the same architecture that consists of eight truss modules. The following demonstrates step by step how the workspace of the 8-module manipulator is generated in accordance with the preceding sections. Each truss module of the manipulator under consideration has the same kinematic parameters: the width of the base and top is $w = 0.2$; each of the three actuated legs can each attain four different states that are equally spaced between $q_{min} = 0.15$ and $q_{max} = 0.25$. The number of configurations that the 8-module manipulator can attain is hence $4^{24} \approx 2.8 \times 10^{14}$. As input we are given the workspace density of the 4-module manipulator in the form of a histogram.

1. The workspace density of the 4-module manipulator is shown in Fig. 6. The dimensions of each pixel are $\Delta x$

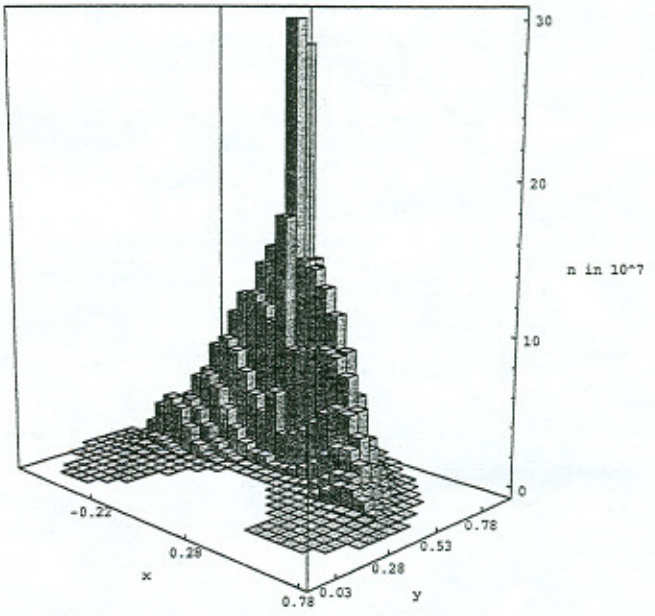


**Fig. 6 Discrete density of manipulator with 4 modules**

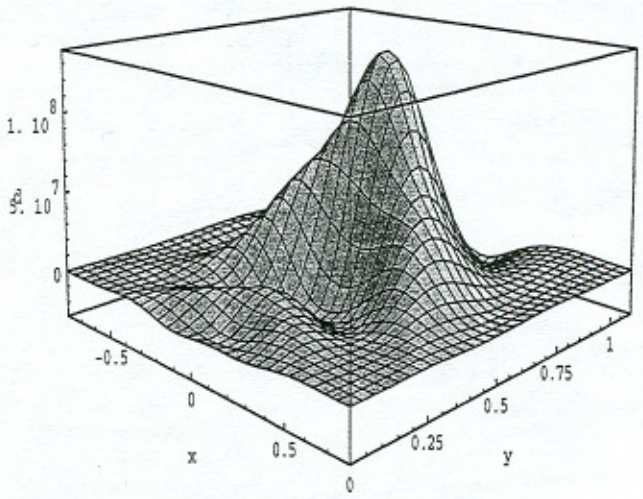


Fig. 7 Hermite-Fourier Series for density of manipulator with 4 modules

$= \Delta y = 0.05$ and $\Delta\theta = 2\pi/50$. To simplify the plotting the data is projected in the $x$, $y$-plane by summing over all different values of the angular variable $\theta$, and the density is plotted as height over $(x, y)$.

2. The fitting of a Hermite series to the histogram is shown in Fig. 7, using $N = 10$ terms in each of the directions $x$, $y$ and $\theta$. Shown again is the projection to the $x$, $y$-plane which in this case results from integration over all orientation angles. The scale factor in the series approximation is chosen as $L = 0.25$.
3. The Hermite series is convolved with itself (since the 8-module manipulator consists of two identical 4-module parts), and the result is shown in Fig. 8. This approximation has $3N$ terms in $x$- and $y$-direction and $N$ terms in $\theta$-direction.
4. The number of terms in the series approximation is reduced to the number used before convolution ($N \times N \times N$). The result is shown in Fig. 9. The appearance of ripples is typical for Hermite series approximations of very low order.
5. To allow a comparison with other methods we also included the result obtained from a discrete convolution as described in Chirikjian and Ebert-Uphoff (1998). That particular implementation was not designed to deal with very high densities, so that densities above a certain threshold are cut. This occurs before the results are projected to the $x$, $y$-plane and in effect restricts the values in Fig. 10 to be smaller then $10^{13}$. However, Fig. 10 serves as a reference for the shape of the workspace in Figs. 8 and 9.

It has to be considered that the time for each convolution, although independent of the number of modules considered, depends on the number of coefficients being used. As mentioned in Section 3.2.4, each convolution is of order $\mathbf{O}(N^{10})$ which limits the number of coefficients $N$ that can be used in the approximation. However, once $N$ is fixed, each convolution takes a constant amount of time and can be repeated any number of times to generate the workspace of manipulators with 16 modules, 32 modules, etc.

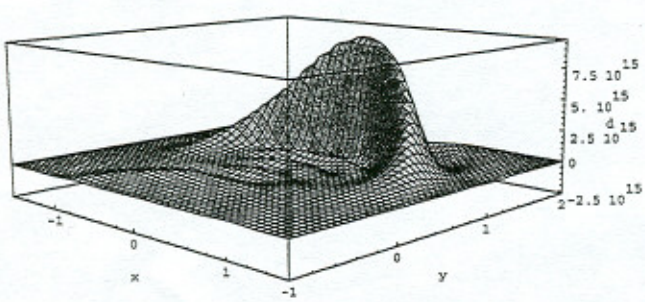


Fig. 8 Convolution of Hermite-Fourier Series (density of 8 modules)

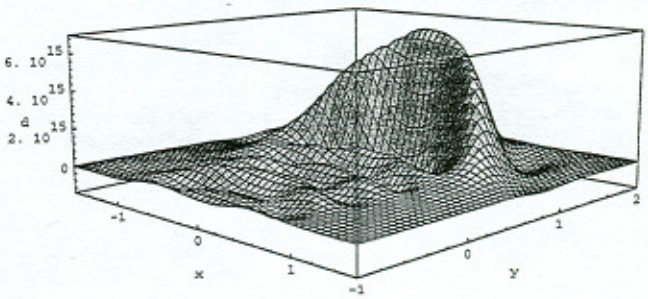


Fig. 9 Convolution of Hermite-Fourier Series cut down to 10 terms

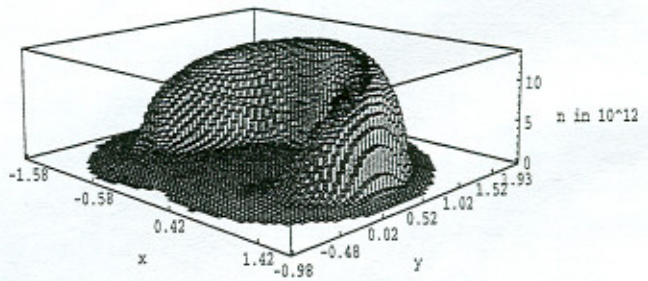


Fig. 10 Discrete density of manipulator with 8 modules

The algorithm was implemented on a SUN SPARCstation 5, 110 MHz, in the C programming language. Figures were made using Mathematica version 2.1. The running time to generate the workspace density for four modules by brute force is about five minutes. To perform the convolution takes two hours. Hence the running time to generate the workspace density for eight modules following the above procedure is about two hours, while brute force would have required more than a decade.

## 5 Discussion and Conclusions

In this paper the concept of closed-form convolution of functions on $SE(D)$ is explained and applied to the efficient generation of discretely actuated planar manipulator workspace density for the case when $D = 2$. Explicit closed-form approximation of workspace densities are obtained using the Hermite-Fourier series. It is shown that this series approximation captures the qualitative shape of discrete data, and allows for efficient convolution.

For a manipulator which is a cascade of $B$ identical discretely-actuated modules with $K$ states each, this reduces numerical computations from $O(K^B)$ to $\mathbf{O}(N^{10} \log B)$, where $N^3$ is the number of functions in the series approximation. It remains to refine these techniques, to generalize them to the three dimensional case, and to numerically solve the mathematical inverse problem of manipulator design for desired density.

## References

Alciatore, D., and Ng, C.-C. D., 1994, "Determining Manipulator Workspace Boundaries using the Monte Carlo Method and Least Squares Segmentation," *ASME Robotics: Kinematics, Dynamics and Controls*, DE-72:141–146.

Blackmore, D., and Leu, M., 1992, "Analysis of Swept Volume via Lie Groups and Differential Equations," *The International Journal of Robotics Research*, Vol. 11, No. 6, pp. 516–537.

Ceccarelli, M., and Vinciguerra, A., 1995, "On the Workspace of General 4R Manipulators," *The International Journal of Robotics Research,* Vol. 14, No. 2, pp. 152–160.

Chirikjian, G., 1996, "Fredholm Integral Equations on the Euclidean Motion Group," *Inverse Problems,* Vol. 12, pp. 579–599.

Chirikjian, G., and Ebert-Uphoff, I., 1998, "Numerical Convolution on the Euclidean Group with Applications to Workspace Generation," *IEEE Trans. on Robotics and Automation,* Vol. 14, No. 1, pp. 123–136.

Dwarakanath, T., Ghosal, A., and Shrinivasa, U., 1994, "Kinematic Analysis of Articulated Design of Articulated Manipulators with Joint Motion Constraints," ASME JOURNAL OF MECHANICAL DESIGN, Vol. 116.

Ebert-Uphoff, I., 1997, "On the Development of Discretely-Actuated Hybrid-Serial-Parallel Manipulators," Ph.D. thesis, The Johns Hopkins University, Dept. of Mechanical Engineering.

Ebert-Uphoff, I., and Chirikjian, G., 1995, "Efficient Workspace Generation for Binary Manipulators with Many Actuators," *Journal of Robotic Systems,* Vol. 12, No. 6, pp. 383–400.

Ebert-Uphoff, I., and Chirikjian, G., 1996a, "Inverse Kinematics of Discretely Actuated Hyper-Redundant Manipulators Using Workspace Densities," *In Proceedings 1996 IEEE International Conference on Robotics and Automation,* pp. 139–145.

Ebert-Uphoff, I., and Chirikjian, G., 1996b, "Discretely Actuated Manipulator Workspace Generation by Closed Form Convolution," *1996 ASME Mechanisms Conference Proceedings,* Irvine, California.

Emiris, D., 1993, "Workspace Analysis of Realistic Elbow and Dual-Elbow Robot," *Mechanisms and Machine Theory,* Vol. 28, No. 3, pp. 375–396.

Gosselin, C., and Guillot, M., 1991, "The Synthesis of Manipulators With Prescribed Workspace," ASME JOURNAL OF MECHANICAL DESIGN, Vol. 113, No. 4, pp. 451–455.

Gupta, K., and Roth, B., 1982, "Design Considerations for Manipulator Workspace," ASME JOURNAL OF MECHANICAL DESIGN, Vol. 104:704–711.

Koliskor, A., 1986, "The 1-Coordinate Approach to the Industrial Robots Design," In *Information Control Problems in Manufacturing Technology 1986. Proceedings of the 5th IFAC/IFIP/IMACS/IFORS Conference,* pp. 225–232, Suzdal, USSR (Preprint).

Korein, J., 1985, *A Geometric Investigation of Reach,* MIT Press.

Kumar, A., and Waldron, K. J., 1981a, "Numerical Plotting of Surfaces of Positioning Accuracy of Manipulators," *Mech. Mach. Theory,* Vol. 16, No. 4, pp. 361–368.

Kumar, A., and Waldron, K. J., 1981b, "The Workspace of a Mechanical Manipulator," ASME JOURNAL OF MECHANICAL DESIGN, Vol. 103, pp. 665–672.

Kumar, V., 1992, "Characterization of Workspaces of Parallel Manipulators," ASME JOURNAL OF MECHANICAL DESIGN, Vol. 114, pp. 368–375.

Kwon, S.-J., Youm, Y., and Chung, W., 1994, "General Algorithm for Automatic Generation of the Workspace for $n$-link Planar Redundant Manipulators," ASME JOURNAL OF MECHANICAL DESIGN, Vol. 116, pp. 967–969.

Lai, Z.-C., and Menq, C.-H., 1988, "The Dextrous Workspace of Simple Manipulators," *IEEE Journal of Robotics and Automation,* Vol. 4, No. 1, pp. 99–103.

Merlet, J.-P., 1992, "Manipulateurs Paralleles, 5eme Partie: Determination de l'espace de Travail a Orientation Constante," *Technical Report 1645, INRIA,* Sophia Antipolis.

Merlet, J.-P., 1995, "Determination of the Orientation Workspace of Parallel Manipulators," *Journal of Intelligent and Robotic Systems,* Vol. 13, pp. 143–160.

Murray, R., Li, Z., and Sastry, S., 1994, *A Mathematical Introduction to Robotic Manipulation,* CRC Press.

Park, F., and Brockett, R., 1994, "Kinematic Dexterity of Robotic Mechanisms," *The International Journal of Robotics Research,* Vol. 13, No. 1, pp. 1–15.

Pieper, D., 1968, "The Kinematics of Manipulators under Computer Control," Ph.D. thesis, Stanford University.

Rastegar, J., and Deravi, P., 1987, "Methods to Determine Workspace, its Subspaces With Different Numbers of Configurations and All the Possible Configurations of a Manipulator," *Mech. Mach. Theory,* Vol. 22, No. 4, pp. 343–350.

Roth, B., Rastegar, J., and Scheinman, V., 1973, "On the Design of Computer Controlled Manipulators," *In First CISM-IFTMM Symp. on Theory and Practice of Robots and Manipulators,* pp. 93–113 (see pp. 106–108).

Selfridge, R. G., 1983, "The Reachable Workarea of a Manipulator," *Mech. Mach. Theory,* Vol. 18, No. 2, pp. 131–137.

Sen, D., and Mruthyunjaya, T., 1994, "A Discrete State Perspective of Manipulator Workspaces," *Mech. Mach. Theory,* Vol. 29, No. 4, pp. 591–605.

Tsai, Y., and Soni, A., 1981, "Accessible Region and Synthesis of Robot Arms," ASME JOURNAL OF MECHANICAL DESIGN, Vol. 103, pp. 803–811.

Vilenkin, N., and Klimyk, A., 1991, *Representation of Lie Groups and Special Functions,* Kluwer Academic Publ.

Yang, D., and Lee, T., 1983, "On the Workspace of Mechanical Manipulators," ASME JOURNAL OF MECHANISMS, TRANSMISSIONS, AND AUTOMATION IN DESIGN, Vol. 105, pp. 62–69.