Analysis of Deterministic Sampling Sequences for Sampling-based Path Planners

Máximo A. ROA[1,*], and Jan ROSELL[2]

[1,2]Institute of Industrial and Control Engineering (IOC)
Technical University of Catalonia (UPC)
E-mail: (maximo.roa[1], jan.rosell[2])@upc.edu

***Abstract*—Deterministic sampling has been demonstrated to be competitive with respect to probabilistic sampling in sampling-based path planners. This paper proposes a simple and yet efficient deterministic sampling sequence for the sampling of a d-dimensional $\mathcal{C}$-space. The sequence is based on the recursive use, over a multi-grid cell decomposition, of the ordering of the 2^d descendant cells of any parent cell. This ordering is generated by the digital construction method using a $d \times d$ matrix T_d. The paper proposes different general expressions of this matrix and contributes with a comparative study of the different resulting sequences over several $\mathcal{C}$-spaces: $\Re^3$, $SO(3)$ and $\Re^6$. The performance index used for the comparison is the mutual distance, i.e. the minimum distance (measured with the corresponding metrics) of each new generated sample to all the previous samples of the sequence. The best of the alternatives found can provide good performance results when applied to sampling-based path planners.**

***Index Terms*— Deterministic sampling, sampling-based methods, path planning.**

I. INTRODUCTION

The sampling-based path planners are giving very good results in problems with many degrees of freedom. Their success is mainly due to its sampling-based nature, since the planning paradigm used does not require the difficult explicit characterization of the obstacles of the $\mathcal{C}$-space, but only the interconnection and collision evaluation of a discrete set of sample configurations. Therefore, the generation of samples is one of the crucial factors in the performance of sampling-based planers.

Random sampling is usually used by sampling-based path planners, like the Probabilistic Roadmap Methods (PRMs) [1], or those based on the Rapidly-exploring Random Trees (RRT) [2]. For difficult path-planning problems, like those involving narrow passages, the number of samples required by a uniform sampling might be quite large. This is the reason why several importance sampling methods have been introduced (e.g. [3], [4]). Those strategies increase the density of sampling in some given areas of the $\mathcal{C}$-space, thus making easier to find a solution using a reasonable amount of samples.

In comparison to those approaches, deterministic sampling sequences have been proposed [5]. These deterministic sampling sequences have the advantages of classical grid search approaches, i.e. a lattice structure (that allows to easily determine the neighborhood relations) and a good uniform coverage of the $\mathcal{C}$-space. Deterministic sampling sequences applied to PRM-like planners are demonstrated to achieve good results compared to the basic PRM planner [6]. Moreover, they can be used in importance sampling strategies since these strategies usually rely on a uniform generation of samples that are then filtered by different criteria to select samples in critical regions.

This paper has as objective the proposal of a general deterministic sampling sequence and the execution of a comparative study over several $\mathcal{C}$-spaces between different variants. The paper is structured as follows. Section II introduces the deterministic sampling sequence that is computed on a multi-grid cell decomposition of a parameter space, and based on the recursive application of an ordering L_d of the 2^d descendant cells of any parent cell. Section III discusses the mapping of the parameter space to the $\mathcal{C}$-space and Section IV proposes different ways to generate L_d. Then, a comparative study of the different proposed variants of the sequence over several $\mathcal{C}$-spaces ($\Re^3$, $SO(3)$ and $\Re^6$) is performed in Section V. Finally, Section VI concludes the work.

II. DETERMINISTIC SAMPLING SEQUENCE

The deterministic sampling sequence used is based on a multi-grid cell decomposition, introduced in Section II-A, and on the low-dispersion ordering of the descendant cells of any given parent cell, introduced in Section II-B. The expression of the sequence is given in Section II-C.

*Departament of Mechanical and Mechatronic Engineering - National University of Colombia; at present with the IOC-UPC, supported by the Programme Alβan (European Union Programme of High Level Scholarships for Latin America), Scholarship No. E04D039103CO.

This work was partially supported by the CICYT projects DPI2004-03104 and DPI2005-00112

A. Multi-grid cell decomposition of $\mathcal{C}$-space

A multi-grid decomposition of a d-dimensional $\mathcal{C}$-space is considered. An initial cell with sides of unitary size covering the entire $\mathcal{C}$-space composes the first grid. The levels in the multi-grid are called partition levels and are enumerated such that the first one is level 0 and the maximum resolution corresponds to partition level M (partition levels are denoted by super-indices). A cell of a given partition level M is called a sample or an M-cell; and is coded as follows. Let:

- The index matrix V^M be the binary $d \times M$ matrix whose rows are the binary representation of the indices $v_j^M \ \ \forall j \in 1 \dots d$ of an M-cell on the regular grid of partition level M:

$$V^M = \begin{pmatrix} v_1^M \\ \vdots \\ v_j^M \\ \vdots \\ v_d^M \end{pmatrix} = \begin{pmatrix} a_{M1} & \dots & a_{i1} & \dots & a_{11} \\ \vdots & & \vdots & & \vdots \\ a_{Mj} & \dots & a_{ij} & \dots & a_{1j} \\ \vdots & & \vdots & & \vdots \\ a_{Md} & \dots & a_{id} & \dots & a_{1d} \end{pmatrix} \tag{1}$$

  being a_{Mj} and a_{1j} the most and least significant bit, respectively, of the binary representation of v_j^M.
- W^M be a $d \times M$ matrix of weights, with:

$$w_{ij} = 2^{(M-j)d+i-1} \text{ for } i \in 1 \dots d \ \ , j \in 1 \dots M \tag{2}$$

Then, the sample code C^M and its index matrix V^M are related as follows:

$$C^M = V^M \cdot W^M \tag{3}$$

$$V^M = C^M \& W^M \tag{4}$$

where the operation $A \cdot B$ represents the scalar product of matrices A and B, and the operation $a \& B$ between a scalar a and a matrix B computes the bit-AND operation between a and all the components b_{ij} of B. As an example the conversion operations of cell code 22 with indices (6,1) on the grid of partition level $M = 3$ (Figure 1a) are:

$$C^3 = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 16 & 4 & 1 \\ 32 & 8 & 2 \end{pmatrix} = 22 \tag{5}$$

$$V^3 = 22 \& \begin{pmatrix} 16 & 4 & 1 \\ 32 & 8 & 2 \end{pmatrix} = $$

$$= 010110 \& \begin{pmatrix} 010000 & 000100 & 000001 \\ 100000 & 001000 & 000010 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{6}$$

The cell code of any m-cell, with $m < M$, is made coincident with the code of the first M-cell it contains (i.e. the descendant M-cell with lower cell code), as illustrated in Figure 1b.

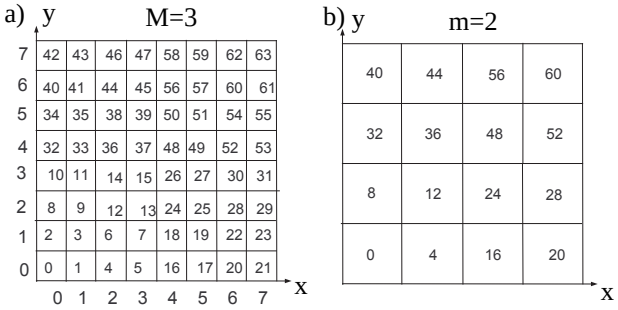

Fig. 1. Coding of M-cells (a) and of 2-cells (b) when $M = 3$.

B. Ordering of descendant cells

The position of a cell with respect to its parent cell can be defined by a binary word, i, with d bits, one for each axis, i.e. $i = \sum_{j=1}^{d} n_j 2^{j-1}$.

Finding a low-dispersion ordering of the 2^d descendant cells of a parent cell is then equivalent to the finding of the sequence, L_d, of 2^d binary words such that each element of the sequence maximizes the mutual distance, i.e. the minimum distance to the previous elements of the sequence. This criterion is further discussed in Section V-A.

L_d can be obtained using a digital construction method [7] that finds the sequence multiplying a $d \times d$ binary matrix, T_d, by the binary representation of the indices of the sequence:

$$L_d(i) = T_d \ \ i = T_d \begin{pmatrix} n_1 \\ \vdots \\ n_d \end{pmatrix} \tag{7}$$

Different proposals for T_d are introduced in Section IV and the comparative study of their effects on the sequence is detailed in Section V. An example of T_d for $d = 2$ and $d = 3$ is:

$$T_2 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \qquad T_3 = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \tag{8}$$

C. The sampling sequence

The sampling sequence, $s_d(k)$, is a sequence of sample codes that specifies the ordering in which the d-dimensional $\mathcal{C}$-space is to be explored. The sequence $s_d(k)$ is based on the recursive use of L_d.

Let $k \geq 0$ be the index of the sequence and T_d be the matrix that determines the cell ordering of the descendant cells as introduced in the previous section. Then:

$$s_d(k) = (T_d V_k^M) \cdot W'^M \tag{9}$$

TABLE I
FIRST 20 SAMPLES OF SEQUENCE s_2 AND FIRST 10 SAMPLES OF r_2^{48}.

k	0	1	2	3	4	5	6	7	8	9
$s_2[k]$	0	48	32	16	12	60	44	28	8	56
k	10	11	12	13	14	15	16	17	18	19
$s_2[k]$	40	24	4	52	36	20	3	51	35	19
k	0	1	2	3	4	5	6	7	8	9
$r_2^{48}[k]$	48	60	56	52	51	63	59	55	50	62

where V_k^M is the index matrix corresponding to k, the product $T_d V_k^M$ is the standard binary matrix multiplication between matrices T_d and V_k^M, and W'^M is a $d \times M$ matrix of weights, with:

$$w'_{ij} = 2^{(j-1)d+i-1} \text{ for } i \in 1 \dots d \;\; j \in 1 \dots M \quad (10)$$

(Note that matrix W'^M coincides with W^M if the order of its columns is exchanged).

If only the samples of a given cell of $\mathcal{C}$-space are necessary, they can be obtained with the following (re)sampling sequence. Let m_K be the partition level of that cell and K be its code. Then:

$$r_d^K(j) = K + (T_d V_j^{(M-m_K)}) \cdot W'^{(M-m_K)} \text{ with } j \geq 1 \quad (11)$$

As an example, considering $M = 3$ and the expression of T_2 proposed in (8), the sample corresponding to $k = 6$ is:

$$\begin{aligned} s_2(6) &= \left[\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \right] \cdot \begin{pmatrix} 1 & 4 & 16 \\ 2 & 8 & 32 \end{pmatrix} \\ &= \begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 4 & 16 \\ 2 & 8 & 32 \end{pmatrix} = 44 \end{aligned} \quad (12)$$

and the sample generated by $r_2(k)$ over the 1-cell 48 (i.e. the top right corner of the $\mathcal{C}$-space) for $k = 6$ is:

$$\begin{aligned} r_2^{48}(6) &= 48 + \left[\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right] \cdot \begin{pmatrix} 1 & 4 \\ 2 & 8 \end{pmatrix} \\ &= 48 + \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 4 \\ 2 & 8 \end{pmatrix} = 48 + 11 = 59 \end{aligned} \quad (13)$$

Finally, the first 20 samples generated by $s_2(k)$ and the first 10 generated by $r_2^{48}(k)$ are shown in Table I. Following these sequences over Figure 1a gives a good understanding of how they incrementally and uniformly cover the space.

III. MAPPING $\mathcal{C}$-SPACES

The deterministic sampling sequence proposed in the previous section is used to obtain samples of a d-dimensional unit cube of parameters, that must be appropriately mapped to configurations of a particular $\mathcal{C}$-space.

A. Manipulators

For the path planning of robot manipulators of d d.o.f., the $\mathcal{C}$-space is $\Re^d$. Taking into account the maximum displacement of each joint, the configuration coordinates can be scaled to the unit cube $[0,1]^d \subset \Re^d$. Then, the indices $(v_1^M, \dots, v_d^M)$ on the grid of partition level M of the samples obtained by the deterministic sampling sequence are mapped to coordinates of the $\mathcal{C}$-space as follows:

$$x_j = v_j^M s_M + \frac{s_M}{2} \qquad \forall j \in 1 \dots d \quad (14)$$

being s_M the size of the sides of the M-cells.

B. Rigid body motions

For the path planning of a 3D rigid-body that can both translate and rotate the $\mathcal{C}$-space is $SE(3)$, the Special Euclidean Group in three dimensions. This group is the cartesian product of the Euclidean space $\Re^3$ with the Special Orthogonal Group $SO(3)$ of 3×3 orthonormal matrices that represent, respectively, arbitrary translations and rotations in 3 dimensions [8].

Following the main ideas given in [9], the mapping of the indices $(v_1^M, \dots, v_6^M)$ of the samples obtained by the deterministic sampling sequence $s_6(d)$ are mapped to configurations of $SE(3)$ as follows.

1) Translations: The position of a 3D rigid body is defined by the coordinates of the origin of its reference frame with respect to a reference frame fixed at the workspace. Taking into account the maximum displacement in each direction, the coordinates can be scaled to the unit cube $[0,1]^3 \subset \Re^3$, and the mapping of the indices (v_1^M, v_2^M, v_3^M) to the coordinates (x_1, x_2, x_3) is done as in (14).

2) Rotations: The orientation of a 3D rigid body can be defined by a rotation direction and a rotation vector. For the rotation direction, the indices v_4^M and v_5^M are used to parameterize the hierarchical triangular decomposition of the surface of a tetrahedron inscribed in the unit sphere. For the rotation angle, the index v_6^M is used as:

$$\theta = 2 \arccos(v_6^M s_M + \frac{s_M}{2}) \quad (15)$$

This is a nonlinear expression that differs from the linear one proposed in [9] (i.e. $\theta = v_6^M \pi$) and provides a more uniform mapping, because of the space topology, as measured by a standard bi-invariant metrics (Section V-A).

IV. Ordering Alternatives

In the digital construction method introduced in [7] the matrix T_d is proposed to be built by columns in an incremental way, but no general expression or procedure is given except for the first column that must be composed of ones in order to place the second sample as far as possible of the first one. The only general expression proposed until now is that of [10], that generates an ever non-increasing mutual distance for the elements of L_d although the maximization of the mutual distance (Section V-A) is not guaranteed.

This section discusses three proposals of T_d in order to compare its performance in the generation of the deterministic sampling sequence. All three proposals satisfy the main constraint of having a general expression for any dimension d of the $\mathcal{C}$-space.

A. Proposal T_d^A

This matrix is the one proposed in [10] and it will be used as a base of comparison for the other proposals, since it is the only general expression proposed for T_d until now. It is constructed as follows. Each column $j \in 1 \dots d$ is composed of $(j-1)$ zeros followed by a 1 (that corresponds to the diagonal). The rest of the column is filled by alternating $(j-1)$ zeros with $(j-1)$ ones until the column is completed. The resulting T_d is a lower diagonal matrix with non-zero diagonal elements, and therefore is full rank. As a consequence, T_d is able to generate all the 2^d elements of the sequence.

As an example, for $d = 10$ the matrix is:

$$T_{10}^A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (16)$$

B. Proposal T_d^B

This matrix satisfies $T_d = T_d^{-1}$, i.e. the index in the sequence of a given sample can be obtained by using the same expression s_d. This feature can provide a more efficient coding of the algorithms used in sampling-based planners. Its construction is based on a recursive use of T_2^B for any d that is a power of 2:

$$T_2^B = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \quad (17)$$

$$T_4^B = \begin{pmatrix} T_2^B & 0 \\ T_2^B & T_2^B \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix} \quad (18)$$

$$T_8^B = \begin{pmatrix} T_4^B & 0 \\ T_4^B & T_4^B \end{pmatrix} = \begin{pmatrix} T_2^B & 0 & 0 & 0 \\ T_2^B & T_2^B & 0 & 0 \\ T_2^B & 0 & T_2^B & 0 \\ T_2^B & T_2^B & T_2^B & T_2^B \end{pmatrix} \quad (19)$$

When d is not a power of 2 then:

$$T_d^B = \text{Trunc}_d(T_D^B) \quad (20)$$

with D being the smallest power of 2 satisfying $D > d$ and $\text{Trunc}_d(T_D)$ being the square matrix composed of the first d rows and columns of T_D.

As an example, T_3^B and T_6^B are:

$$T_3^B = \text{Trunc}_3(T_4^B) = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \quad (21)$$

$$T_6^B = \text{Trunc}_6(T_8^B) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 \end{pmatrix} \quad (22)$$

C. Proposal T_d^C

This matrix is based on a prime decomposition. For each prime dimension, i.e. $d = 2, 3, 5, 7, \dots$, T_d is first defined:

$$T_2^C = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \quad (23)$$

$$T_3^C = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \quad (24)$$

$$T_d^C = \text{Trunc}_d(T_{(d+1)}^C) \;\; \forall d \text{ prime s.t. } d \geq 5 \quad (25)$$

Then, for any d, a recursive construction is done based on the prime decomposition of d, e.g.:

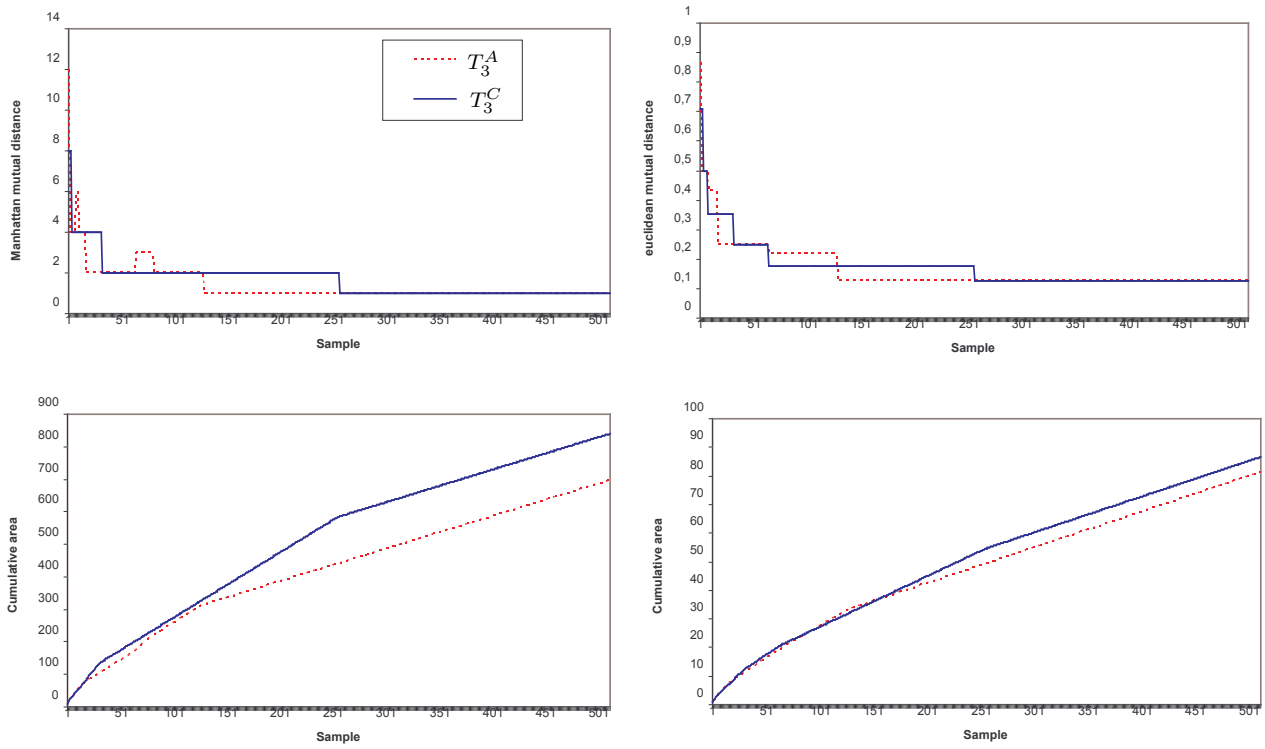

Fig. 2. Top: Mutual distances (Manhattan (left) and Euclidean (right)) in the sampling of $\Re^3$ with $M = 3$; Bottom: Cumulative area in the sampling of $\Re^3$ (Manhattan (left) and Euclidean (right)).

$$T_6^C = \begin{pmatrix} T_3^C & 0 \\ T_3^C & T_3^C \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \end{pmatrix} \quad (26)$$

$$T_9^C = \begin{pmatrix} T_3^C & T_3^C & 0 \\ 0 & T_3 & 0 \\ T_3^C & 0 & T_3^C \end{pmatrix} \quad (27)$$

It can be seen that this approach results with a first column not composed exclusively of ones. This makes that the second sample is not as far as possible from the first one, i.e. the second sample does not maximize the mutual distance. Nevertheless, as shown in the next section, in the long run it results in a better performance, i.e. in a slower decreasing of the mutual distance.

V. Comparative Study

A. Performance index

Different uniformity measures have been proposed in the literature [6]. Perhaps the most useful in path planning is dispersion, a metrics-based criterion. Let $X = [0, 1]^d \subset \Re^d$ be the space where to generate samples. Let P be the set of samples taken from X. Then, dispersion for P is defined as:

$$\delta(P, \rho) = \sup_{q \in X} \min_{p \in P} \rho(q, p) \quad (28)$$

being ρ any metrics on X. Intuitively, it corresponds to the largest empty ball inscribed in X and with the center in X. A criterion that emphasizes dispersion requires that points be as far away from each other as possible, e.g. the mutual distance of the samples must be maximized [7]. Mutual distance on P is defined as:

$$\rho_m(P) = \min_{x,y \in P} \rho(x, y) \quad (29)$$

In this paper the mutual distance will be used as the basic performance index, using the following metrics:

- Manhattan distance between the cells codes in the parameters space.
- Euclidean metrics between the samples in the $\mathcal{C}$-space, in the case of $\Re^d$.
- Standard bi-invariant metrics between the samples in the $\mathcal{C}$-space, in the case of $SO(3)$. Being R_1 and R_2 two elements in $SO(3)$, this metrics is [11]:

$$d(R_1, R_2) = \left\|\log\left(R_1^{-1}R_2\right)\right\| = \arccos\left(\frac{\text{Trace}\left(R_1^{-1}R_2\right) - 1}{2}\right) \quad (30)$$

However, a more useful way to compare the performance of the ordering alternatives is the comparison of the cumulative area (i.e. the integral) of the mutual distance. This shows how far the samples are being placed from each other as a progression in the sample number, visualizing the gain of one method compared to another.

B. Sampling in $\Re^3$

The first experiment is the comparison of the use of T_3^C with $T_3^A = T_3^B$ in the sampling of $\Re^3$. Using $M = 3$, Figure 2 (top) shows the Manhattan mutual distance (left) and the mutual distance measured with Euclidean metrics (right) computed for all the 512 samples of the sequence. The continuous blue line corresponds to T_3^C and the red point line to T_3^A. It can be seen in both figures that continuous line decreases more slowly showing that this approach gets a better uniform covering of the space as samples are placed. The cumulative area of the mutual distance is shown in Figure 2 (bottom); as stated previously, T_3^C provides a better performance.

C. Sampling in $SO(3)$

The second experiment is the comparison of the use of T_3^C with $T_3^A = T_3^B$ in the sampling of $SO(3)$. Using $M = 3$, Figure 3 (top) shows the mutual distance measured with the standard bi-invariant metrics for this space, while Figure 3 (bottom) shows the cumulative area of the previous one. Manhattan mutual distance results are the same that those obtained for $\Re^3$, as both spaces use T_3 for sampling. Differences between the two building methods are not obvious using bi-invariant metrics mutual distance, i.e. in the sampling of $SO(3)$ the performance obtained using alternative matrices T_d is roughly the same.

D. Sampling in $\Re^6$

The last experiment is the comparison of the use of T_6^C with T_6^A and T_6^B in the sampling of $\Re^6$. Using $M = 1$, Figure 4 (top) shows the Manhattan and the Euclidean mutual distances for all the 64 samples of the sequence. Figure 4 (bottom) shows the cumulative area for both measures of mutual distance. Continuous blue line corresponds to T_6^C, dashed green line corresponds to T_6^B and dotted red line corresponds to T_6^A. It can be seen that in this case, like in $\Re^3$, T_d^C also provides the best uniform coverage of $\mathcal{C}$-space.

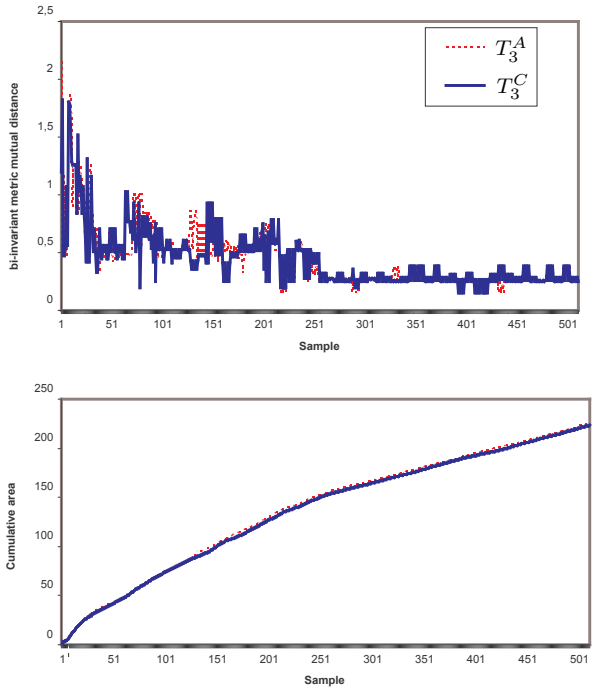

Fig. 3. Top: Bi-invariant metrics mutual distance in the sampling of $SO(3)$, with $M = 3$; Bottom: Bi-invariant metrics cumulative area in the sampling of $SO(3)$.

VI. Conclusions

The performance of sampling-based path planners relies on the set of samples used. For uniform sampling, deterministic sampling sequences are a good alternative since they outperform probabilistic methods in terms of the dispersion obtained.

This paper proposes a simple and yet efficient deterministic sampling sequence computed on a multi-grid decomposition of a parameter space and the corresponding mapping to the $\mathcal{C}$-space. The paper contributes with a comparative study of the performance of different variants of the sampling sequence over the $\mathcal{C}$-spaces $\Re^3$, $SO(3)$ and $\Re^6$.

Each variant depends on the choice of T_d, the $d \times d$ matrix used to generate the ordering L_d of the 2^d descendant cells of any parent cell. The results of the comparative study show that:

- The criterium of maximizing the mutual distance at each new sample of L_d, proposed in [7], does not guarantee the best performance in terms of uniform covering; e.g. placing the second sample as far as possible from the first one greatly constrains where the next samples should be placed, making them to have a lower mutual distance than expected.
- An useful way to compare the performance of the

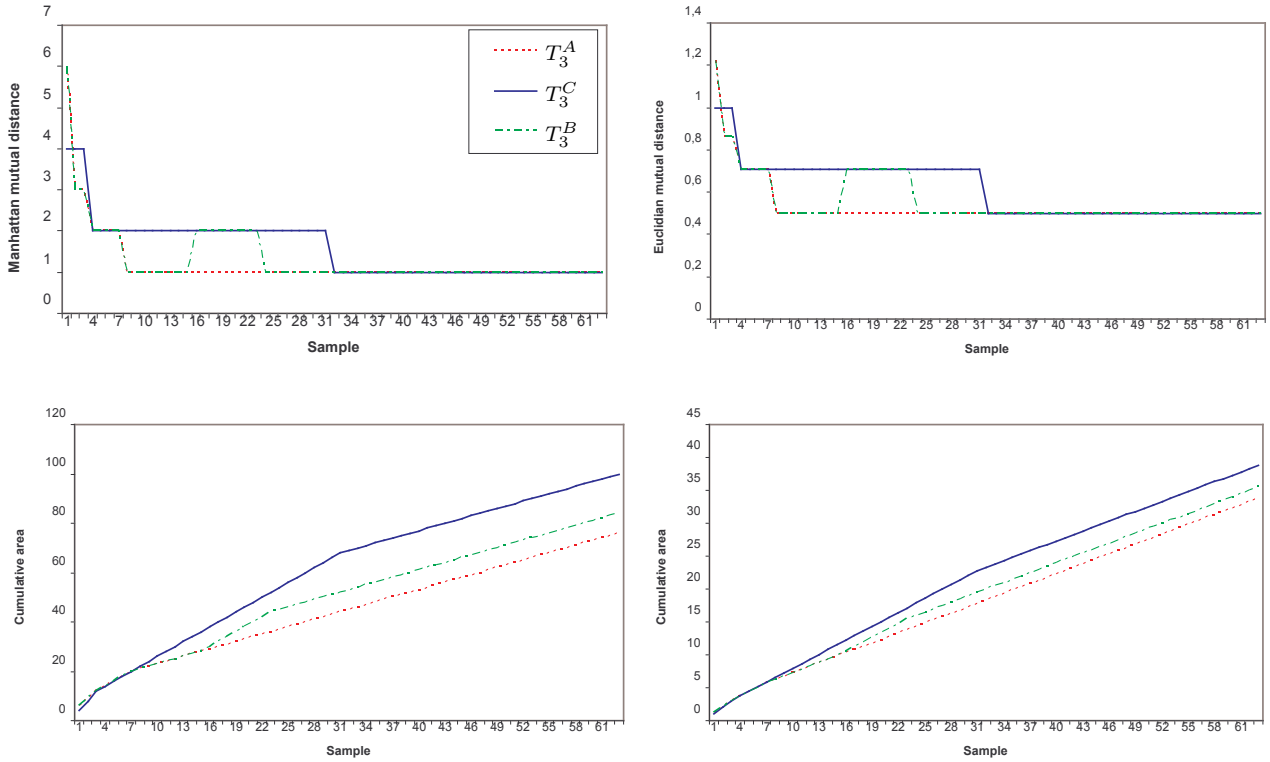

Fig. 4. Top: Mutual distances (Manhattan (left) and Euclidean (right)) in the sampling of $\Re^6$ with $M = 1$; Bottom: Cumulative area in the sampling of $\Re^6$ (Manhattan (left) and Euclidean (right)).

ordering alternatives is the comparison of the cumulative area of the mutual distance. This shows how far the samples are being placed from each other as a progression in the sample number, visualizing the gain of one method compared to another. Under this performance measure, the alternative T_d^C provided the best sampling sequence and is the one proposed to be applied to sampling-based path planners.

- The performance of a sequence depends on the $\mathcal{C}$-space where it is used, since the mutual distance must be measured in the appropriate metrics. In the case of $SO(3)$ differences between the different variants proposed resulted not significant.

Future work includes the implementation of the proposed deterministic sequence T_d^C in a PRM planner to prove its performance in a challenging test, e.g. the bend-corridor problem with different degrees of freedom.

REFERENCES

[1] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. K. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. volume 12, pages 566–580, August 1996.

[2] J. J. Kuffner and S. M. LaValle. RRT-connect: An efficient approach to single-query path planning. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 995–1001, 2000.

[3] P. Leven and S. Hutchinson. Using manipulability to bias sampling during the construction of probabilistic roadmaps. *IEEE Trans. on Robotics and Automation*, 19(6):1020–1026, 2003.

[4] D. Hsu, T. Jiang, J. Reif, and Z. Sun. The bridge test for sampling narrow passages with probabilistic roadmap planners. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 4420–4426, 2003.

[5] M. S. Branicky, S. M. LaValle, K. Olson, and L. Yang. Quasi-randomized path planning. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 1481–1487, 2001.

[6] S. M. LaValle, M. S. Branicky, and S. R. Lindemann. On the relationship between classical grid search and probabilistic roadmaps. *Int. J. of Robotics Res.*, 23(7-8):673–692, 2004.

[7] S. R. Lindemann, A. Yershova, and S. M. LaValle. Incremental grid sampling strategies in robotics. In *Proc. of the Sixth Int. Workshop on the Algorithmic Foundations of Robotics*, 2004.

[8] J.-C. Latombe. *Robot Motion Planning*, chapter Configuration Space of a Rigid Object, pages 58–104. Kluwer Academic Publishers, 1991.

[9] J. Rosell. Sampling SE(3) with a deterministic sequence for 3D rigid-body path planning. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 2154–2159, 2005.

[10] J. Rosell and M. Heisse. An efficient deterministic sequence for sampling-based motion planners. In *Proc. of the IEEE Int. Symp. on Assembly and Task Planning*, 2005.

[11] F. C. Park. Distance metrics on the rigid-body motions with applications to mechanical design. *Journal of Mechanical Design*, 117:48–54, 1995.