Journal of Complexity 32 (2016) 74–80

Contents lists available at ScienceDirect

Journal of Complexity

journal homepage: www.elsevier.com/locate/jco

On a projection-corrected component-by-component construction

Josef Dick [a], Peter Kritzer [b,*]

[a] School of Mathematics and Statistics, The University of New South Wales, Sydney, 2052 NSW, Australia

[b] Department of Financial Mathematics and Applied Number Theory, Johannes Kepler University Linz, Altenbergerstr. 69, 4040 Linz, Austria

ARTICLE INFO

Article history:
Received 3 February 2015
Accepted 4 August 2015
Available online 20 August 2015

Keywords:
Lattice point sets
Polynomial lattice point sets
Component-by-component algorithm

ABSTRACT

The component-by-component construction is the standard method of finding good lattice rules or polynomial lattice rules for numerical integration. Several authors have reported that in numerical experiments the generating vector sometimes has repeated components. We study a variation of the classical component-by-component algorithm for the construction of lattice or polynomial lattice point sets where the components are forced to differ from each other. This avoids the problem of having projections where all quadrature points lie on the main diagonal. Since the previous results on the worst-case error do not apply to this modified algorithm, we prove such an error bound here. We also discuss further restrictions on the choice of components in the component-by-component algorithm.

Introduction

Lattice point sets are integration node sets frequently used in quasi-Monte Carlo rules

$$\frac{1}{N}\sum_{n=0}^{N-1} f(\boldsymbol{x}_n) \approx \int_{[0,1]^s} f(\boldsymbol{x})\, \mathrm{d}\boldsymbol{x}$$

* Corresponding author.
E-mail addresses: josef.dick@unsw.edu.au (J. Dick), peter.kritzer@jku.at (P. Kritzer).

http://dx.doi.org/10.1016/j.jco.2015.08.001
0885-064X/

for the approximation of s-dimensional integrals over the unit cube $[0,1]^s$. For a modulus N (N a positive integer) and a generating vector $\boldsymbol{g}=(g_1,\dots,g_s)\in\{1,2,\dots,N-1\}^s$, a (rank one) lattice point set is an integration node set of the form

$$\boldsymbol{x}_n=\left(\left\{\frac{ng_1}{N}\right\},\dots,\left\{\frac{ng_s}{N}\right\}\right),\quad n=0,1,\dots,N-1.$$

Here, for real numbers $x\geq 0$ we write $\{x\}=x-\lfloor x\rfloor$ for the fractional part of x. For vectors $\boldsymbol{x}$ we apply $\{\cdot\}$ component-wise. A quasi-Monte Carlo rule using a lattice point set is called lattice rule. For further information on lattice rules we refer to [5,15,20].

We consider a weighted Korobov space with general weights as studied in [8,16]. Before we do so we need to introduce some notation. Let $\mathbb{Z}$ be the set of integers and let $\mathbb{Z}_*=\mathbb{Z}\setminus\{0\}$. Furthermore, $\mathbb{N}$ denotes the set of positive integers. For a set $\mathcal{E}$ we denote by $|\mathcal{E}|$ the cardinality of $\mathcal{E}$. For $s\in\mathbb{N}$ we write $[s]=\{1,2,\dots,s\}$. For a vector $\boldsymbol{x}=(x_1,\dots,x_s)\in[0,1]^s$ and for $u\subseteq[s]$ we write $\boldsymbol{x}_u=(x_j)_{j\in u}\in[0,1]^{|u|}$ and $(\boldsymbol{x}_u,\boldsymbol{0})\in[0,1]^s$ for the vector $(y_1,\dots,y_s)$ with $y_j=x_j$ if $j\in u$ and $y_j=0$ if $j\notin u$.

The importance of the different components or groups of components of the functions from the Korobov space to be defined is specified by a sequence of positive weights $\boldsymbol{\gamma}=(\gamma_u)_{u\subseteq[s]}$, see [21], where we may assume that $\gamma_\emptyset=1$. The smoothness is described by a parameter $\alpha>1$. The weighted Korobov space $\mathcal{H}(K_{s,\alpha,\boldsymbol{\gamma}})$ is a reproducing kernel Hilbert space with kernel function of the form

$$K_{s,\alpha,\boldsymbol{\gamma}}(\boldsymbol{x},\boldsymbol{y})=1+\sum_{\emptyset\neq u\subseteq[s]}\gamma_u\sum_{\boldsymbol{h}_u\in\mathbb{Z}_*^{|u|}}\frac{\exp(2\pi\mathtt{i}\boldsymbol{h}_u\cdot(\boldsymbol{x}_u-\boldsymbol{y}_u))}{\prod_{j\in u}|h_j|^\alpha}.$$

It is well known in the theory of lattice rules that it is useful to restrict the range of a generating vector $\boldsymbol{g}$ of an N-point lattice point set to $\mathcal{Z}_N^s$, where

$$\mathcal{Z}_N=\{k\in\{1,2,\dots,N-1\}:\gcd(k,N)=1\}.$$

Furthermore, it is known (see, for example, [8]) that the squared worst-case error of a lattice rule generated by a generating vector $\boldsymbol{g}\in\mathcal{Z}_N^s$ in the weighted Korobov space $\mathcal{H}(K_{s,\alpha,\boldsymbol{\gamma}})$ is given by

$$e^2(\boldsymbol{g})=\sum_{\substack{\boldsymbol{h}\in\mathbb{Z}^s\setminus\{\boldsymbol{0}\}\\ \boldsymbol{g}\cdot\boldsymbol{h}\equiv 0 \pmod N}} r_\alpha(\boldsymbol{\gamma},\boldsymbol{h}),\tag{1}$$

where for $\emptyset\neq u\subseteq[s]$ and $\boldsymbol{h}_u\in\mathbb{Z}_*^{|u|}$ we have

$$r(\boldsymbol{\gamma},(\boldsymbol{h}_u,\boldsymbol{0}))=\gamma_u\prod_{j\in u}|h_j|^{-\alpha}.$$

It is known that the worst-case error in the Korobov space coincides with the worst-case error in the unanchored Sobolev space using the tent-transform [6] and is also related to the mean square worst-case error for randomly shifted lattice rules [5]. Hence the results here automatically also apply to those cases.

The result

The now standard method for finding good generating vectors for numerical integration in Korobov spaces is the so-called component-by-component (CBC) construction (see [1,12]). We can set the first component to 1 and then proceed inductively by choosing one new component at a time by minimizing the error criterion $e^2(g_1^*,g_2^*,\dots,g_{d-1}^*,g)$ as a function of the last (not yet fixed) component $g\in\mathcal{Z}_N$. Here, the components $g_1^*,g_2^*,\dots,g_{d-1}^*$ have been fixed in the previous steps.

It has been observed that in running the CBC construction it may happen that components repeat themselves, i.e., there are $i,j\in\{1,\dots,s\}$, $i\neq j$, such that $g_i^*=g_j^*$. We quote from [13]:

> *[…] However, it has been observed that the components start to repeat from some dimension onward for product-type weights, hence leading to a practical limit on the value of d [we remark that d has the role of s in [13]]. This side effect of the CBC algorithm is yet to be fully understood.*