



**HAL**  
open science

## Blue Noise through Optimal Transport

Fernando de Goes, Katherine Breeden, Victor Ostromoukhov, Mathieu  
Desbrun

► **To cite this version:**

Fernando de Goes, Katherine Breeden, Victor Ostromoukhov, Mathieu Desbrun. Blue Noise through Optimal Transport. ACM Transactions on Graphics, Association for Computing Machinery, 2012, 6, 31, pp.171:1-171:1. 10.1145/2366145.2366190 . hal-01353135

**HAL Id: hal-01353135**

**<https://hal.archives-ouvertes.fr/hal-01353135>**

Submitted on 7 Mar 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Blue Noise through Optimal Transport

Fernando de Goes  
Caltech

Katherine Breeden  
Stanford

Victor Ostromoukhov  
Lyon 1 U./CNRS

Mathieu Desbrun  
Caltech

## Abstract

We present a fast, scalable algorithm to generate high-quality blue noise point distributions of arbitrary density functions. At its core is a novel formulation of the recently-introduced concept of capacity-constrained Voronoi tessellation as an optimal transport problem. This insight leads to a continuous formulation able to enforce the capacity constraints exactly, unlike previous work. We exploit the variational nature of this formulation to design an efficient optimization technique of point distributions via constrained minimization in the space of power diagrams. Our mathematical, algorithmic, and practical contributions lead to high-quality blue noise point sets with improved spectral and spatial properties.

**CR Categories:** I.4.1 [Image Processing and Computer Vision]: Digitization and Image Capture—Sampling.

**Keywords:** Blue noise, power diagram, capacity constraints.

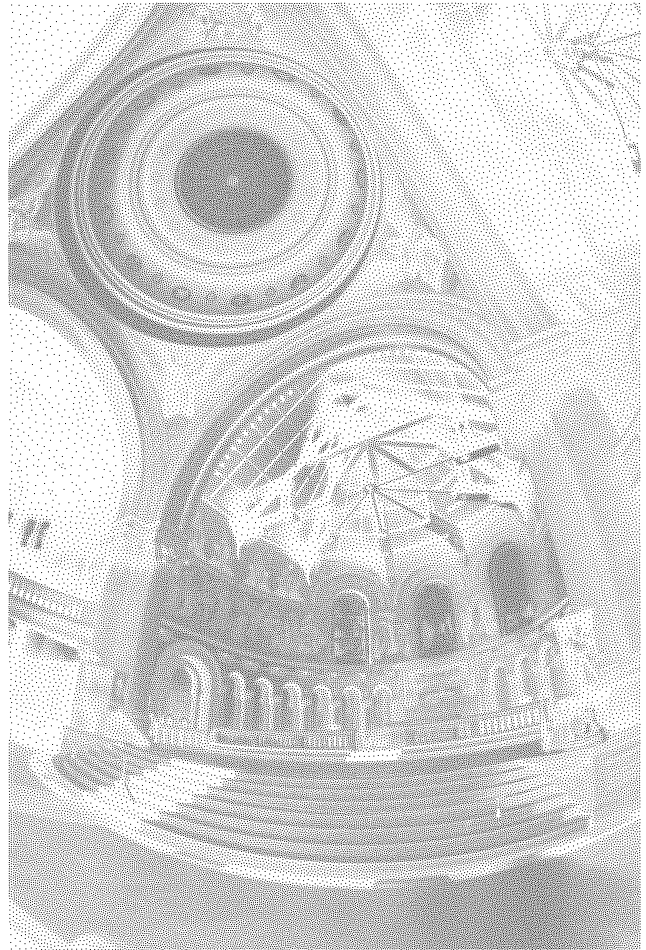
**Links:** [DL](#) [PDF](#) [WEB](#) [CODE](#)

## 1 Introduction

Coined by Ulichney [1987], the term *blue noise* refers to an even, isotropic, yet unstructured distribution of points. Blue noise was first recognized as crucial in dithering of images since it captures the intensity of an image through its local point density, without introducing artificial structures of its own. It rapidly became prevalent in various scientific fields, especially in computer graphics, where its isotropic properties lead to high-quality sampling of multidimensional signals, and its absence of structure prevents aliasing. It has even been argued that its visual efficacy (used to some extent in stippling and pointillism) is linked to the presence of a blue-noise arrangement of photoreceptors in the retina [Yellott 1983].

### 1.1 Previous Work

Over the years, a variety of research efforts targeting both the characteristics and the generation of blue noise distributions have been conducted in graphics. Arguably the oldest approach to algorithmically generate point distributions with a good balance between density control and spatial irregularity is through error diffusion [Floyd and Steinberg 1976; Ulichney 1987], which is particularly well adapted to low-level hardware implementation in printers. Concurrently, a keen interest in uniform, regularity-free distributions appeared in computer rendering in the context of anti-aliasing [Crow 1977]. Cook [1986] proposed the first dart-throwing algorithm to create Poisson disk distributions, for which no two points are closer together than a certain threshold. Considerable efforts followed to modify and improve this original algorithm [Mitchell 1987; McCool and Fiume 1992; Jones 2006; Bridson 2007; Gamito and Maddock 2009]. Today’s best Poisson disc algorithms are very efficient and versatile [Dunbar and Humphreys 2006; Ebeida et al. 2011], even



**Figure 1: Memorial.** Our variational approach allows sampling of arbitrary functions (e.g., a high-dynamic range image courtesy of P. Debevec), producing high-quality, detail-capturing blue noise point distributions without spurious regular patterns (100K points, 498 s).

running on GPUs [Wei 2008; Bowers et al. 2010; Xiang et al. 2011]. Fast generation of irregular low-discrepancy sequences have also been proposed [Niederreiter 1992; Lemieux 2009]; however, these methods based on the radical-inverse function rarely generate high-quality blue noise.

In an effort to allow fast blue noise generation, the idea of using patterns computed offline was raised in [Dippé and Wold 1985]. To remove potential aliasing artifacts due to repeated patterns, Cohen et al. [2003] recommended the use of non-periodic Wang tiles, which subsequently led to improved hierarchical sampling [Kopf et al. 2006] and a series of other tile-based alternatives [Ostromoukhov et al. 2004; Lagae and Dutré 2006; Ostromoukhov 2007]. However, all precalculated structures used in this family of approaches rely on the offline generation of high-quality blue noise.

Consequently, a number of researchers focused on developing methods to compute point sets with high-quality blue noise properties, typically by evenly distributing points over a domain via Lloyd-based iterations [McCool and Fiume 1992; Deussen et al. 2000; Secord 2002; Balzer et al. 2009; Xu et al. 2011; Chen et al. 2012], electro-

static forces [Schmaltz et al. 2010], statistical-mechanics interacting Gaussian particle models [Fattal 2011], or farthest-point optimization [Schlömer et al. 2011]. These iterative methods consistently generate much improved point distributions, albeit at sometimes excessive computational complexity.

Finally, recent efforts have provided tools to analyze point sets using spatial/spectral [Lagae and Dutré 2008; Schlömer and Deussen 2011] and differential [Wei and Wang 2011] methods. Extensions to anisotropic [Li et al. 2010b; Xu et al. 2012], non-uniform [Wei and Wang 2011], multiclass [Wei 2010], and general spectrum sampling [Zhou et al. 2012] have also been recently introduced.

## 1.2 Motivation and Rationale

Despite typically being slower, optimization methods based on iterative displacements of points have consistently been proven superior to other blue noise generation techniques. With the exception of [Schmaltz et al. 2010; Fattal 2011], these iterative approaches rely on Voronoi diagrams and Lloyd’s relaxations [Lloyd 1982]. To our knowledge, the use of Lloyd’s algorithm for blue noise sampling was first advocated in [McCool and Fiume 1992] to distribute points by minimizing the root mean square (RMS) error of the quantization of a probability distribution. However, the authors noticed that a “somewhat suboptimal solution” was desirable to avoid periodic distribution: Lloyd’s algorithm run to convergence tends to generate regular regions with point or curve defects, creating visual artifacts. Hence, a limited number of iterations was used in practice until Balzer et al. [2009] proposed the use of a Capacity-Constrained Voronoi Tessellation (CCVT), a rather drastic change in which a constraint of equi-area partitioning is added to algorithmically ensure that each point conveys equal visual importance. However, this original approach and its various improvements rely on a discretization of the capacities, and thus suffer from a quadratic complexity, rendering even GPU implementations [Li et al. 2010a] unable to gracefully scale up to large point sets. Two variants were recently proposed to improve performance, both providing an approximation of CCVT by penalizing the area variance of either Voronoi cells [Chen et al. 2012] or Delaunay triangles [Xu et al. 2011].

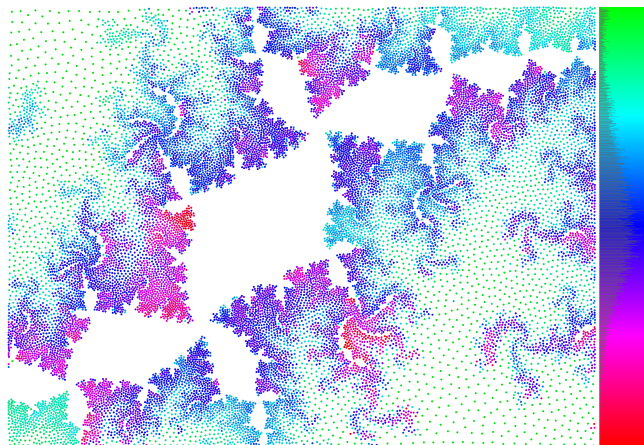
## 1.3 Contributions

In this paper, we show that CCVT can be formulated as a constrained optimal transport problem. This insight leads to a continuous formulation able to enforce the capacity constraints exactly, unlike related work. The variational nature of our formulation is also amenable to a fast, scalable, and reliable numerical treatment. Our resulting algorithm will be shown, through spectral analysis and comparisons, to generate high-grade blue noise distributions. Key differences from previous methods include:

- a reformulation of CCVT as a continuous constrained minimization based on optimal transport, as opposed to the discretized approximation suggested in [Balzer and Heck 2008];
- an optimization procedure over the space of *power diagrams* that satisfies the capacity constraints up to numerical precision, as opposed to an approximate capacity enforcement in the space of Delaunay triangulations [Xu et al. 2011] or Voronoi diagrams [Chen et al. 2012];
- a regularity-breaking procedure to prevent local aliasing artifacts that occur in previous approaches.

## 2 Redefining Blue Noise through Optimal Transport

Before presenting our algorithm for point set generation, we spell out our definition of blue noise as a constrained transport problem. We consider an arbitrary domain  $\mathcal{D}$  over which a piecewise-continuous positive field  $\rho$  (e.g., intensity of an image) is defined.



**Figure 2: Fractal.** Optimal transport based blue noise sampling of a Julia set image (20K points). Colors of dots indicate (normalized) weight values, ranging from -30% to 188% of the average squared edge length in the regular triangulation. The histogram of the weights is also shown on top of the color ramp.

## 2.1 Background

Two crucial geometric notions will be needed. We briefly review them next for completeness.

**Optimal Transport.** The optimal transport problem, dating back to Gaspard Monge [Villani 2009], amounts to determining the optimal way to move a pile of sand to a hole of the same volume—where “optimal” means that the integral of the distances by which the sand is moved (one infinitesimal unit of volume at a time) is minimized. The minimum “cost” of moving the piled-up sand to the hole, i.e., the amount of sand that needs to be moved times the  $L_p$  distance it has to be moved, is called the  $p$ -Wasserstein metric. The 2-Wasserstein metric, using the  $L_2$  norm, is most common, and is often referred to as the earth mover’s distance. Optimal transport has recently been of interest in many scientific fields; see [Mullen et al. 2011; Bonneel et al. 2011; de Goes et al. 2011] for graphics applications.

**Power Diagrams.** From a point set  $\mathbf{X} = \{\mathbf{x}_i\}_{i=1\dots n}$  a natural partition of a domain  $\mathcal{D}$  can be obtained by assigning every location in  $\mathcal{D}$  to its nearest point  $\mathbf{x}_i \in \mathbf{X}$ . The region  $\mathcal{V}_i$  assigned to point  $\mathbf{x}_i$  is known as its Voronoi region, and the set of all these regions forms a partition called the Voronoi diagram. While this geometric structure (and its dual, the Delaunay triangulation of the point set) has found countless applications, power diagrams offer an even more general way to partition a domain based on a point set. They involve the notion of a *weighted* point set, defined as a pair  $(\mathbf{X}, W) = \{(\mathbf{x}_1, w_1), \dots, (\mathbf{x}_n, w_n)\}$ , where  $\mathbf{X}$  is a set of points and  $W = \{w_i\}_{i=1\dots n}$  are real numbers called weights. The *power distance* from a position  $\mathbf{x}$  to a weighted point  $(\mathbf{x}_i, w_i)$  is defined as  $\|\mathbf{x} - \mathbf{x}_i\|^2 - w_i$ , where  $\|\cdot\|$  indicates the Euclidean distance. Using this definition, with each  $\mathbf{x}_i$  we associate a power cell (also called weighted Voronoi region)

$$\mathcal{V}_i^w = \{\mathbf{x} \in \mathcal{D} \mid \|\mathbf{x} - \mathbf{x}_i\|^2 - w_i \leq \|\mathbf{x} - \mathbf{x}_j\|^2 - w_j, \forall j\}.$$

The power diagram of  $(\mathbf{X}, W)$  is the cell complex formed by the power cells  $\mathcal{V}_i^w$ . Note that when the weights are all equal, the power diagram coincides with the Voronoi diagram of  $\mathbf{X}$ ; power diagrams and their associated dual (called regular triangulations) thus generalize the usual Voronoi/Delaunay duality.

## 2.2 Blue Noise as a Constrained Transport Problem

Sampling a density function  $\rho(\mathbf{x})$  consists of picking a few representative points  $\mathbf{x}_i$  that capture  $\rho$  well. This is, in essence, the halftoning process that a black-and-white printer or a monochrome pointillist painter uses to represent an image. In order to formally



**Figure 3: Zebra.** Since our approach accurately captures variations of density, we can blue-noise sample images containing both fuzzy and sharp edges (160K-pixel original image (top right) courtesy of Frédo Durand). 40K points, generated in 159 seconds.

characterize a blue noise distribution of points, we see sampling as the process of *aggregating*  $n$  disjoint regions  $\mathcal{V}_i$  (forming a partition  $\mathcal{V}$  of the domain  $\mathcal{D}$ ) into  $n$  points  $\mathbf{x}_i$ : if  $\rho$  is seen as a density of ink over  $\mathcal{D}$ , sampling consists in coalescing this distribution of ink into  $n$  Dirac functions (i.e., ink dots).

We can now revisit the definition of blue noise sampling through the following requirements:

**A. Uniform Sampling:** all point samples should equally contribute to capturing the field  $\rho$ . Consequently, their associated regions  $\mathcal{V}_i$  must all represent the same amount  $m$  of ink:

$$m_i = \int_{\mathcal{V}_i} \rho(\mathbf{x}) \, d\mathbf{x} \equiv m.$$

**B. Optimal Transport:** the total cost of transporting ink from the distribution  $\rho$  to the finite point set  $\mathbf{X}$  should be minimized, thus representing the most effective aggregation. This ink transport cost for an arbitrary partition  $\mathcal{V}$  is given as

$$\mathcal{E}(\mathbf{X}, \mathcal{V}) = \sum_i \int_{\mathcal{V}_i} \rho(\mathbf{x}) \|\mathbf{x} - \mathbf{x}_i\|^2 \, d\mathbf{x},$$

i.e., as the sum per region of the integral of all displacements of the local ink distribution  $\rho$  to its associated ink dot.

**C. Local Irregularity:** the point set should be void of visual artifacts such as Moiré patterns and other aliasing effects; that is, it should be free of local spatial regularity.

Note that the first requirement implies that the resulting local point density will be *proportional to*  $\rho$  as often required in importance sampling. The second requirement favors *isotropic* distribution of points since such partitions minimize the transport cost. The final requirement prevents regular or hexagonal grid patterns from emerging. Together, these three requirements provide a *density-adapted, isotropic, yet unstructured distribution of points*, capturing the essence of a blue noise as a constrained transport problem.

### 2.3 Power Diagrams vs. Voronoi Diagrams

While the cost  $\mathcal{E}$  may resemble the well-known CVT energy [Du et al. 1999], the reader will notice that it is more general, as the cells  $\mathcal{V}_i$  are *not* restricted to be Voronoi. In fact, Aurenhammer et al. [1998] proved that capacity constrained partitions (requirement A) that minimize the cost  $\mathcal{E}$  (requirement B) for a given point set are power diagrams. So, instead of searching through the entire space of possible partitions, we can rather restrict partitions  $\mathcal{V}$  to be power diagrams, that is,  $\mathcal{V}_i \equiv \mathcal{V}_i^w$ . Within this subspace of partitions, the cost functional  $\mathcal{E}$  coincides with the  $\star^0$ -HOT<sub>2,2</sub> energy of [Mullen

et al. 2011] (i.e., the power diagram version of the CVT energy). This difference is crucial: while methods restricting their search to Delaunay meshes [Xu et al. 2011] or Voronoi diagrams [Chen et al. 2012] can only approximate the constraints in requirement A, this power diagram formulation has the additional variables (weights) necessary to allow *exact* constraint enforcement, thus capturing sharp feature much more clearly than previous methods (see Sec. 5). In fact, all of our results exhibit uneven weights as demonstrated in Fig. 2, reinforcing the importance of power vs. Voronoi diagrams.

## 3 Variational Formulation

Leveraging the fact that requirements A and B can only be enforced for power diagrams, we describe next our variational characterization of blue noise distributions of weighted point sets  $(\mathbf{X}, W)$ . Requirement C will be enforced algorithmically, as discussed in Sec. 4.6, by detecting regularity and locally jittering the point set to guide our optimization towards non-regular distributions.

### 3.1 Functional Extremization

We can now properly formulate our constrained minimization to enforce requirements A and B.

**Lagrangian formulation.** A common approach to deal with a constrained minimization is to use Lagrange multipliers  $\Lambda = \{\lambda_i\}_{i=1 \dots n}$  to enforce the  $n$  constraints (one per point) induced by requirement A. The resulting optimization procedure can be stated as:

$$\text{Extremize } \mathcal{E}(\mathbf{X}, W) + \sum_i \lambda_i (m_i - m)$$

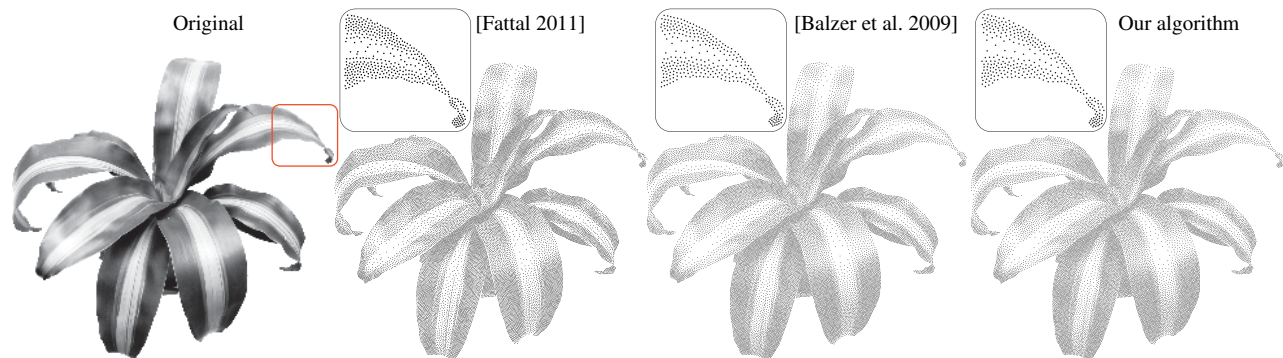
with respect to  $\mathbf{x}_i$ ,  $w_i$ , and  $\lambda_i$ , where the functional  $\mathcal{E}$  is now clearly labeled with the point set and its weights as input (since we know that only power diagrams can optimize the constrained transport energy), and  $m_i$  is the amount of ink in the region  $\mathcal{V}_i^w$ :

$$\mathcal{E}(\mathbf{X}, W) = \sum_i \int_{\mathcal{V}_i^w} \rho(\mathbf{x}) \|\mathbf{x} - \mathbf{x}_i\|^2 \, d\mathbf{x}, \quad m_i = \int_{\mathcal{V}_i^w} \rho(\mathbf{x}) \, d\mathbf{x}. \quad (1)$$

**Simpler formulation.** The Lagrangian multipliers add undue complexity: they contribute an additional  $n$  variables to the optimization. Instead, one can extremize a simpler function  $\mathcal{F}$  depending only on the weighted point set: we show in the appendix that the extremization above is equivalent to finding a stationary point of the following scalar functional:

$$\mathcal{F}(\mathbf{X}, W) = \mathcal{E}(\mathbf{X}, W) - \sum_i w_i (m_i - m). \quad (2)$$

With  $n$  fewer variables to deal with, we will show in Sec. 4 that blue noise generation can be efficiently achieved.



**Figure 4: Stippling.** Test from [Secord 2002] (20K points). While [Fattal 2011] does not capture density gradients very cleanly (see close-ups), our result is similar to CCVT [Balzer et al. 2009] on this example, at a fraction of the CPU time. Comparative data courtesy of the authors.

### 3.2 Functional Properties

The closed-form expression of our functional allows us not only to justify the Lloyd-based algorithmic approaches previously used in [Balzer and Heck 2008; Balzer et al. 2009; Li et al. 2010a], but also to derive better numerical methods to find blue noise point sets by exploiting a few key properties.

$\mathcal{F}(\mathbf{X}, \mathbf{W})$ : For a fixed set of points  $\mathbf{X}$ , the Hessian of our functional w.r.t. weights is the negated *weighted Laplacian operator* as shown in the appendix. Consequently, extremizing  $\mathcal{F}$  is actually a maximization with respect to all  $w_i$ 's. This is an important insight that will lead us to an efficient numerical approach comparable in speed to recent approximate CCVT methods [Xu et al. 2011; Chen et al. 2012], but much faster than the quadratic scheme used in [Balzer and Heck 2008; Balzer et al. 2009; Li et al. 2010a].

$\mathcal{F}(\mathbf{X}, \underline{W})$ : Now for a fixed set of weights  $\underline{W}$ , our functional is the  $\star^0$ -HOT<sub>2,2</sub> energy of [Mullen et al. 2011] (i.e., the power diagram version  $\mathcal{E}$  of the CVT energy), with one extra term due to the constraints. Several numerical methods can be used to minimize this functional. Note that, surprisingly, the functional gradient w.r.t. positions turns out to be simply

$$\nabla_{\mathbf{x}_i} \mathcal{F} = 2m_i(\mathbf{x}_i - \mathbf{b}_i), \quad \text{with } \mathbf{b}_i = \frac{1}{m_i} \int_{V_i^w} \mathbf{x} \rho(\mathbf{x}) d\mathbf{x}, \quad (3)$$

because the boundary term of the Reynolds' transport theorem cancels out the gradients of the constraint terms (see appendix). Extremizing  $\mathcal{F}$  thus implies that we are looking for a “centroidal power diagram”, as  $\mathbf{x}_i$  and its associated weighted barycenter  $\mathbf{b}_i$  have to match to ensure a zero gradient.

### 3.3 Discussion

We now discuss the key differences between our transport-based formulation and previous CCVT methods.

**Discrete vs. Continuous Formulation.** The initial CCVT method and its improvements [Balzer and Heck 2008; Balzer et al. 2009; Li et al. 2010a] adopted a *discrete* formulation in which the density function  $\rho$  is represented by a finite set of samples, with the number of samples being “orders of magnitude” larger than the number  $n$  of points. Blue noise point sets are then generated via repeated energy-decreasing swaps between adjacent clusters, without an explicit use of weights. This discrete setup has several numerical drawbacks. First, while samples can be thought of as quadrature points for capacity evaluation, their use causes accuracy issues: in essence, using samples amounts to quantizing capacities; consequently, the transport part of the CCVT formulation is not strictly minimized. Second, the computational cost induced by the amount of swaps required to reach convergence is quadratic in the number of samples—and thus impractical beyond a few thousand points. Instead, we provided a continuous functional whose extremization formally encodes the concept behind the original CCVT method [Balzer and Heck 2008].

The functional  $\mathcal{F}$  in Eq. 2 was previously introduced in [Aurenhammer et al. 1998] purely as a way to enforce capacity constraints for a fixed point set; here we extend  $\mathcal{F}$  as a function of weights  $w_i$  and positions  $\mathbf{x}_i$ , and the closed-form gradient and Hessian we explicitly derived will permit, in the next section, the development of a fast numerical treatment to generate high-quality blue noise distributions in a scalable fashion, independently of the sampling size of the density function.

**Approximate vs. Exact Constraints.** Attempts at dealing with CCVT through continuous optimization have also been investigated by sacrificing exact enforcement of capacity constraints. In [Balzer et al. 2005; Balzer 2009], for instance, a point-by-point iterative approach is used to minimize the capacity variance of Voronoi cells to best fit the capacity constraints; Chen et al. [2012] recommend adding the capacity variance as a penalty term to the CVT energy instead; Xu et al. [2011] take a dual approach by minimizing capacity variance on Delaunay triangles instead of Voronoi cells. These different variants all mix the requirements of good spatial distribution and capacity constraints into a single minimization, leading to an over-constrained formulation. Minima of their functionals thus always represent a tradeoff between capacity enforcement and isotropic spatial distribution. Instead, our formulation allows exact capacity constraints by controlling the power diagram through the addition of a weight per vertex: we can now optimize distribution quality *while* constraining capacity, resulting in high quality blue noise sampling of arbitrary density field (see quadratic ramp in Fig. 10 for a comparison with recent methods).

## 4 Numerical Optimization

We now delve into the numerical methods and algorithmic details we use to efficiently generate blue noise point distribution based on our variational formulation.

### 4.1 Overall Strategy

We proceed with point set generation by computing a critical point of the functional  $\mathcal{F}$  defined in Eq. 2: we extremize the functional  $\mathcal{F}$  by repeatedly performing a minimization step over positions followed by a projection step over weights to enforce constraints. The power diagram of the weighted point set is updated at each step via the CGAL library [2010]. While this alternating procedure is typical for non-linear multivariable problems, we will benefit from several properties of the functional as already alluded to in Sec. 3:

- enforcing the capacity constraints for a fixed set of point positions is a *concave maximization*;
- minimizing  $\mathcal{F}$  for a fixed set of weights is akin to the minimization of the CVT energy, for which fast methods exist;
- staying clear of regular patterns is enforced algorithmically through a simple local regularity detection and removal.

These three factors conspire to result in a fast and scalable generation of high-quality blue noise point sets as we discuss next.

## 4.2 Constraint Enforcement

For a given set of points  $\mathbf{X}$ , we noted in Sec. 3.1 that finding the set of weights  $W_{\text{opt}}$  to enforce that all capacities are equal is a concave maximization. Fast iterative methods can thus be applied to keep computational complexity to a minimum.

Since the Hessian of  $\mathcal{F}(\mathbf{X}, W)$  is equal to the negated weighted Laplacian  $\Delta^{w,\rho}$  (see appendix), Newton iterations are particularly appropriate to find the optimal set of weights  $W_{\text{opt}}$ . At each iteration, we thus solve the sparse, (Poisson) linear system:

$$\Delta^{w,\rho} \delta = \begin{pmatrix} m-m_1 & m-m_2 & \dots & m-m_n \end{pmatrix}^t, \quad (4)$$

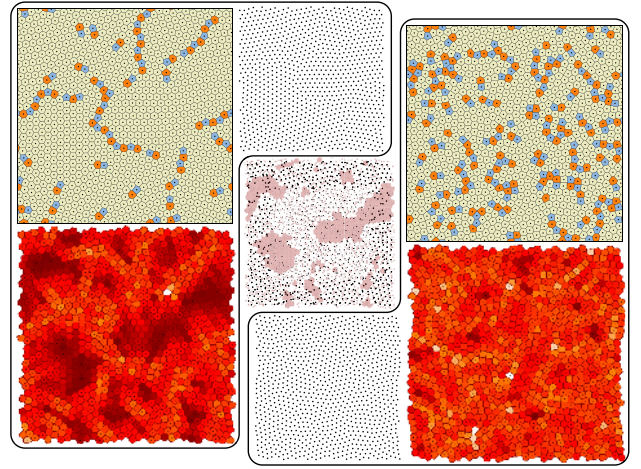
where the righthand side of the equation is equal to the current gradient of  $\mathcal{F}$  w.r.t. weights. A standard line search with Armijo condition [Nocedal and Wright 1999] is then performed to adapt the step size along the vector  $\delta$  before updating the vector  $W$  of current weights. Given that the Hessian is sparse and symmetric, many linear solvers can be used to efficiently solve the linear system used in each Newton iteration; in our implementation, we use the sparse QR factorization method in [Davis 2011]. Typically, it only takes 3 to 5 such iterations to bring the residual of our constraints to within an accuracy of  $10^{-12}$ .

## 4.3 Transport Minimization

For a fixed set of weights  $W$ , we can move the locations of the  $n$  points in order to improve the cost of ink transport  $\mathcal{F}(\mathbf{X}, W)$ . Previous CCVT-based methods [Balzer et al. 2009; Li et al. 2010a] used Lloyd’s algorithm as the method of choice for their discrete optimization. In our continuous optimization context, we have more options. A Lloyd update where positions  $\mathbf{x}_i$  are moved to the barycenter  $\mathbf{b}_i$  of their associated weighted cell  $\mathcal{V}_i^w$  can also be used to reliably decrease the transport cost: indeed, we prove in the appendix that the gradient of  $\mathcal{F}(\mathbf{X}, W)$  is a natural extension of the gradient of the regular CVT energy. However, Lloyd’s algorithm is a special case of a gradient descent that is known to suffer from linear convergence [Du et al. 1999]. We improve the convergence rate through line search, again using adaptive timestep gradient descent with Armijo conditions as proposed in [Mullen et al. 2011]. Note that quasi-Newton iterations as proposed in [Liu et al. 2009] for the CVT energy are not well suited in our context: alternating weight and position optimizations renders the approximation of the Hessian matrix from previous gradients inaccurate, ruining the expected quadratic convergence.

## 4.4 Density Integration

Integrations required by our formulation can be easily handled through quadrature. However, poor quadrature choices may impair the convergence rate of our constraint enforcement. Given that blue noise sampling is most often performed on a rectangular grey-scale image, we design a simple and exact procedure to compute integrals of the density field  $\rho$  inside each cell, as it is relatively inexpensive. Assuming that  $\rho$  is given as a strictly-positive piecewise constant field, we first compute the value  $m$  used in our capacity constraints by simply summing the density values times the area of each constant regions (pixels, typically), divided by  $n$ . We then perform integration within each  $\mathcal{V}_i^w$  in order to obtain the mass  $m_i$ , the barycenter  $\mathbf{b}_i$ , and the individual transport cost for each  $\mathcal{V}_i^w$ . We proceed in three steps. First, we rasterize the edges of the power diagram and find intersections between the image pixels and each edge. Next we perform a scan-line traversal of the image and construct pixel-cell intersections. Integrated densities, barycenters, and transport costs per cell are then accumulated through simple integration within each pixel-cell intersection where the density is constant. Note that our integration differs from previous similar treatments (e.g., [Secord 2002; Lecot and Lévy 2006]) as we provide robust and exact computation not only for cell capacities, but also for their barycenters and transport costs—thus avoiding the need for parameter tweaking required in quadrature approximations.



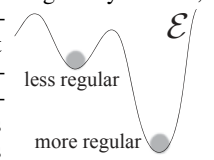
**Figure 5: Breaking Regularity.** Optimization of  $\mathcal{F}$  with a strict convergence threshold ( $\|\nabla_{\mathbf{X}}\mathcal{F}\| \leq 10^{-5}$ ) can produce regularity (left), as revealed by a valence-colored visualization (top) and the distribution of local transport costs  $\mathcal{E}_i$  (bottom). After jittering and relocating aliased regions (middle, colored cells), further optimization brings the point set to a shallower (i.e., less regular) configuration (right) as confirmed by valences and transport costs.

## 4.5 Boundary Treatment

While some of the results we present use a periodic domain (see Sec. 5), most sampling applications involve a bounded domain  $\mathcal{D}$ , often given as a convex polygon (as in the case of a simple image). Dealing with boundaries in our approach is straightforward. First, boundary power cells are clipped by  $\mathcal{D}$  before computing their cell barycenters  $\mathbf{b}_i$  and capacities  $m_i$ . Second, the coefficients of the weighted Laplacian  $\Delta^{w,\rho}$  are computed through the ratio of (possibly clipped) dual edge lengths and primal edge lengths, as proposed in [Mullen et al. 2011]. Thence, the presence of boundaries adds only limited code and computational complexity and it does not affect the convergence rates of any of the steps described above. Note that other boundary treatments could be designed as well, using mirroring or other typical boundary conditions if needed.

## 4.6 Detecting & Breaking Regularities

The numerical procedure described so far solely targets requirements A and B, and as such, nothing preempts regularity. In fact, hexagonal lattices are solutions to our extremization problem in the specific case of constant density and a toroidal domain—and these solutions correspond to “deep” extrema of our functional, as the cost of ink transport  $\mathcal{E}$  reaches a *global* minimum on such regular packings of points. Instead, we algorithmically seek “shallow” extrema to prevent regularity (see inset).



For capacity-constrained configurations, local regularities are easily detected by evaluating the individual terms  $\mathcal{E}_i$  measuring the transport cost within each region  $\mathcal{V}_i^w$ : we assign a regularity score  $r_i$  per point as the local absolute deviation of  $\mathcal{E}_i$ , i.e.,

$$r_i = \frac{1}{|\Omega_i|} \sum_{j \in \Omega_i} |\mathcal{E}_i - \mathcal{E}_j|,$$

where  $\Omega_i$  is the one-ring of  $\mathbf{x}_i$  in the regular triangulation of  $(\mathbf{X}, W)$ . We then refer to the region around a point  $\mathbf{x}_i$  as *aliased* if  $r_i < \tau$ , where the threshold  $\tau = 0.25 m^2$  in all our experiments. When aliased, a point and its immediate neighbors are jittered by a Gaussian noise with a spatial variance of  $1.0/\rho(\mathbf{x}_i)$  and maximum magnitude  $\sqrt{m}$  to break symmetries as recommended in [Lucarini 2009]. To prevent a potential return to the same crystalline configuration during subsequent optimization steps, we further relocate 1% of the

aliased points to introduce defects. Since our numerical approach relies on a line search with Armijo rule (seeking local extrema), starting the optimization from this stochastically scrambled configuration will fall back to a nearby, shallower extremum—hence removing regularity as demonstrated in Fig. 5.

It is worth pointing out that all CVT-based methods (including the existing CCVT schemes) may result in point distributions with local regular patterns. While a few approaches avoided regularity by stopping optimization before convergence, we instead prevent regularity by making sure we stop at shallow minima. This  $\tau$ -based shallowness criterion can be seen as an alternative to the temperature parameter proposed in [Fattal 2011], where the level of excitation of a statistical particle model controls the randomness on the formation of point distributions. Our simple approach is numerically robust and efficient: in practice, we observed that the proposed regularity breaking routine takes place at most once in each example test, independently of the value of  $\tau$ .

#### 4.7 Optimization Schedule

We follow a simple optimization schedule to make the generation process automatic and efficient for arbitrary inputs. We start with a random distribution of points conforming to  $\rho$  (better initialization strategies could be used, of course). We then proceed by systematically alternating optimization of weights (to enforce constraints, Sec. 4.2) and positions (to minimize transport cost, Sec. 4.3). Weight optimization is initialized with zero weights, and iterated until  $\|\nabla_w \mathcal{F}\| \leq 0.1 m$  (the capacity  $m$  is used to properly adapt the convergence threshold to the number of points  $n$  and the density  $\rho$ ). For positions, we optimize our functional until  $\|\nabla_x \mathcal{F}\| \leq 0.1\sqrt{n m^3}$  (again, scaling is chosen here to account for density and number of points). We found that performing Lloyd steps until the gradient norm is below  $0.2\sqrt{n m^3}$  saves computation (it typically requires 5 iterations); only then do we revert to a full-blown adaptive timestep gradient descent until convergence (taking typically 10 iterations). Once an extremum of  $\mathcal{F}$  is found, we apply the regularity detecting-and-breaking procedure presented in Sec. 4.6, and, if an aliased point was found and jittered, we start our optimization again. This simple schedule (see pseudocode in Fig. 6) was used as is on all our results.

## 5 Results

We ran our algorithm on a variety of inputs: from constant density (Fig. 8) to photos (Fig. 1, 3, and 4) and computer-generated images (Fig. 2 and 10), without any need for parameter tuning. Various illustrations based on zoneplates, regularity, and spectral analysis are used throughout the paper to allow easy evaluation of our results and to demonstrate how they compare to previous work.

**Spectral Properties.** The special case of blue noise point distribution for a constant density in a periodic domain has been the subject of countless studies. It is generally accepted that such a point distribution must have a characteristic blue-noise profile for the radial component of its Fourier spectra, as well as low angular anisotropy [Ulichney 1987]. This profile should exhibit no low frequencies (since the density is constant), a high peak around the average distance between adjacent points, along with a flat curve end to guarantee white noise (i.e., no distinguishable features) in the high frequency range. Fig. 8 demonstrates that we improve upon the results of all previous CCVT-related methods, and fare arguably better than alternative methods such as [Fattal 2011]; in particular, we systematically (i.e., not just on average over several distributions, but for every single run) get flat spectrum in low and high frequencies, while keeping high peaks at the characteristic frequency. Note also that the method of Balzer et al. [2009] appears to slowly converge to our results when the ratio  $m/n$  (using their notation) goes to infinity with, evidently, much larger timings (Fig. 7).

---

```

1: // BLUE NOISE THROUGH OPTIMAL TRANSPORT
2: Input: domain  $\mathcal{D}$ , density  $\rho$ , and number of points  $n$ 
3: Initialize  $\mathbf{X}$  with  $n$  random points inside  $\mathcal{D}$  conforming to  $\rho$ 
4: repeat
5:   // Newton method for  $W$ 
6:   Enforce-Capacity-Constraints() (lines 26-33)
7:   // Lloyd step for  $\mathbf{X}$ 
8:    $\mathbf{X} \leftarrow (\mathbf{b}_1 \ \mathbf{b}_2 \ \dots \ \mathbf{b}_n)^t$  (Eq. 3)
9:   Update power diagram
10: until  $\|\nabla_x \mathcal{F}\| \leq 0.2\sqrt{n m^3}$ 
11: repeat
12:   // Newton method for  $W$ 
13:   Enforce-Capacity-Constraints() (lines 26-33)
14:   // Gradient descent for  $\mathbf{X}$ 
15:    $\mathbf{d} = \nabla_x \mathcal{F}$ .
16:   Find  $\beta$  satisfying Armijo condition.
17:    $\mathbf{X} \leftarrow \mathbf{X} - \beta \mathbf{d}$ .
18:   Update power diagram.
19: until  $\|\nabla_x \mathcal{F}\| \leq 0.1\sqrt{n m^3}$ 
20: // Detect and break regularities (Sec. 4.6)
21: if Aliased points detected then
22:   Jitter aliased points and immediate neighbors.
23:   Relocate 1% of aliased points.
24:   Restart optimization (line 4).
25: Output:  $n$  points with blue noise reqs. A, B, and C
26: Subroutine ENFORCE-CAPACITY-CONSTRAINTS()
27: Set  $W = 0$ 
28: repeat
29:   Solve for  $\delta$  in Eq. 4
30:   Find  $\alpha$  satisfying Armijo condition.
31:    $W \leftarrow W + \alpha \delta$ .
32:   Update power diagram
33: until  $\|\nabla_w \mathcal{F}\| \leq 0.1 m$ 

```

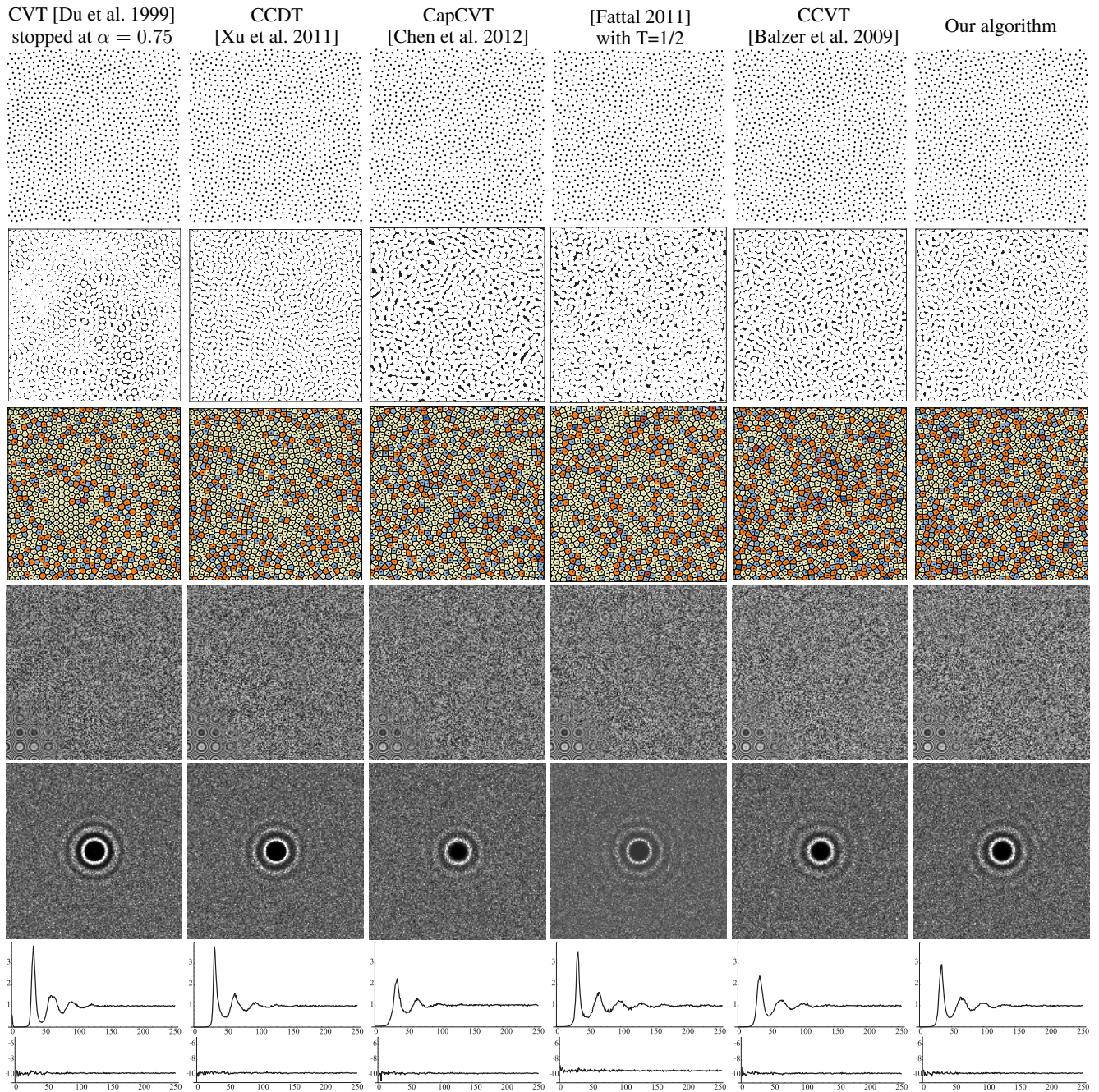
---

**Figure 6:** Pseudocode of the blue noise algorithm.

**Spatial Properties.** We also provide evaluations of the spatial properties of our results. Fig. 8 shows two insightful visualizations of the typical spatial arrangement of our point distributions, side by side with results of previous state-of-the-art methods. The second row shows the gaps between white discs centered on sampling points with a diameter equal to the mean distance between two points; notice the uniformity of gap distribution in our result. The third row compares the number of neighbors for the Voronoi region of each site; as pointed out in [Balzer 2009], the enforcement of the capacity constraints favors heterogenous valences, with fewer noticeable regular regions. Finally, the minimum distance among all points normalized by the radius of a disc in a hexagonal tiling is a measure of distribution quality, known as the normalized Poisson disk radius, and recommended to be in the range  $[0.65, 0.85]$  by [Lagae and Dutré 2008]. In all our constant density blue noise examples, the normalized radius is in the range  $[0.71, 0.76]$ .

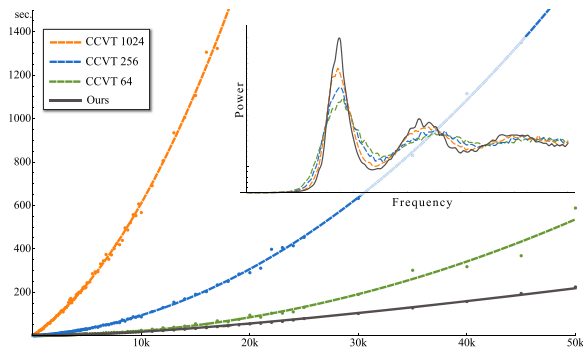
**Quadratic Ramp.** Another common evaluation of blue noise sampling is to generate a point set for an intensity ramp, and count the number of points for each quarter of the ramp. Fig. 10 compares the point sets generated by our technique vs. state-of-the-art methods [Balzer 2009; Fattal 2011; Chen et al. 2012]. While all the methods recover approximately the right counting of points per quarter, our result presents a noticeably less noisy, yet unstructured distribution of points.

**Zoneplates.** We also provide zoneplates in Fig. 8 for the function  $\sin(x^2 + y^2)$ . Each zoneplate image was created via 32x32 copies of



**Figure 8: Comparisons.** Different blue noise algorithms are analyzed for the case of constant density over a periodic domain; Top row: distributions of 1024 points; Second row: gaps between white discs centered on sampling points, over black background. Notice the uniformity of gap distribution in two rightmost point sets. Third row: coloring based on number of neighbors for the Voronoi region of each site; Fourth row:  $1024 \times 1024$  zoneplates for the function  $\sin(x^2 + y^2)$  (see Sec. 5 or [Lagae and Dutré 2006] for details). Fifth row: mean periodograms for 10 independent point sets (except for [Fattal 2011] for which only 5 pointsets were available). Sixth row: radial power spectra—note the pronounced peak in our result, without any increase of regularity. Last row: anisotropy in dB ([Ulichney 1987], p. 56). Data/code for [Fattal 2011] and [Balzer et al. 2009] courtesy of the authors.





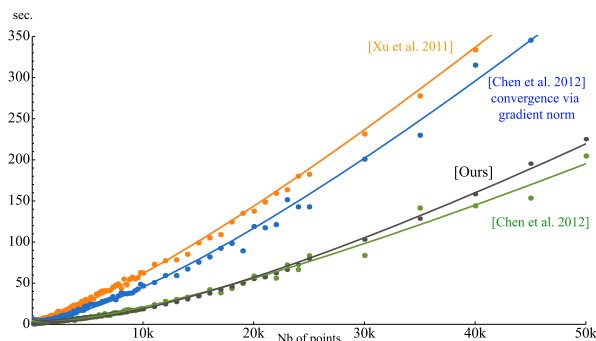
**Figure 7: Discrete vs. Continuous CCVT.** Our timings as a function of the number of points exhibit a typical  $n \log n$  behavior, systematically better than [Balzer et al. 2009]’s  $n^2$ ; yet, our radial spectra (inset, showing averages over 10 runs with 1024 points) even outperforms the fine 1024-sample CCVT results. (Here, CCVT- $X$  stands for  $X$  “points-per-site” as in [Balzer et al. 2009].)

a 1024-point blue noise patch, followed by a Mitchell reconstruction filter to generate a 1024x1024 image with an average of one point per pixel as suggested in [Lagae and Dutré 2006]. Observe the presence of a second noise ring in previous methods, as opposed to the anti-aliased reconstruction achieved by our method.

**Complexity.** Previous CCVT methods analyzed the (worst-case) time complexity of a single iteration of their optimization approach. One iteration of our algorithm involves the construction of a 2D power diagram, costing  $\mathcal{O}(n \log n)$ . It also involves the enforcement of the capacity constraints via a concave maximization w.r.t. the weights via a step-adaptive Newton method; the time complexity of this maximization is of the order of a single Newton step since the convergence rate is quadratic (see [Nocedal and Wright 1999] for a more detailed proof), and therefore incurs the linear cost of solving a sparse (Poisson) linear system. For  $N$ -pixel images and  $n$  points, the total complexity of our algorithm thus becomes  $\mathcal{O}(n \log n + N)$ , with the extra term corresponding to the cost of locating the pixels within each power cell through scan-line traversal. This is significantly better than the discrete versions of CCVT which were either  $\mathcal{O}(n^2 + nN \log N/n)$  [Balzer 2009] or  $\mathcal{O}(n^2 + nN)$  [Li et al. 2010a] and of the same order as the CCVT approximations in [Xu et al. 2011; Chen et al. 2012]. However, we can not match the efficiency of the multi-scale statistical particle model introduced in [Fattal 2011], which scales linearly with the number of points and produces results arguably comparable with the best current methods of blue noise generation.

**Timings.** All of our timings in this paper were clocked on an Intel Core i7 2.2 GHz laptop with 4GB RAM. Depending on the input image resolution and the desired number of points, our approach takes from a few seconds (Figs. 8(right) and 10) to 2.1 minutes for Fig. 4 (20K points, 445x419 image), 2.6 minutes for Fig. 3 (40K points, 600x267 image), 5.8 minutes for Fig. 2 (20K points, 600x450 image), and 8.3 minutes for Fig. 1 (100K points, 512x768 image). The efficiency of the numerics allows us to generate, in 2.8 hours, 1M+ points of the luminance of a high dynamic range 512x768 image (see supplemental material), an order of magnitude more complex than the largest results demonstrated by CCVT-based methods. Note that we purposely developed a code robust to any input and any points-to-pixels ratio. However, code profiling revealed that about 40% of computation time was spent on the exact integration described in Sec. 4.4; depending on the targeted application, performance could thus be easily improved through quadrature [Lecot and Lévy 2006] and/or input image resampling if needed.

**Stopping Criteria.** As discussed in Sec. 4.7, we terminate optimization when  $\|\nabla \mathcal{F}\| < \epsilon$ , i.e., the first order condition for identifying a locally optimal solution to a critical point search [Nocedal and



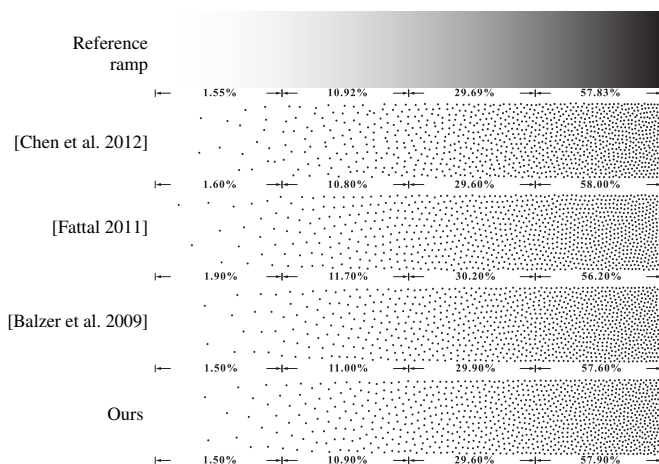
**Figure 9: Performance.** Our method (in grey) performs well despite a stringent convergence criteria ( $\|\nabla \mathcal{F}\| < 0.1 \sqrt{nm^3}$ ). The method of [Chen et al. 2012] (in green) behaves similarly when using a loose stopping criteria based on the functional decrease per iteration; but becomes twice slower (in blue) if the termination is based on the norm of the functional gradient to guarantee local optimality. The code released by [Xu et al. 2011] (in orange) also exhibits comparable performance by terminating the optimization not based on convergence, but after a fixed number of iterations.

Wright 1999]. Recent optimization-based blue noise methods [Xu et al. 2011; Chen et al. 2012], on the other hand, have used the decrease of the objective function per iteration as their stopping criteria. However, a small decrease in the functional does not imply convergence, since a change of functional value depends both on the functional landscape and the step size chosen in each iteration. Favoring guaranteed high quality vs. improved timing, we prefer adopting the first order optimality condition as our termination criteria for robust generation of blue noise distributions. Despite this purposely stringent convergence criteria, the performance of our method is similar to [Chen et al. 2012] with their recommended termination based on functional decrease—but twice faster if the method of [Chen et al. 2012] is modified to use a stricter termination criterion based on the norm of the functional gradient. Xu et al. [2011] advocate a fixed number of iterations, which, again, does not imply either convergence or high-quality results. Our timings and theirs are, however, similar for the type of examples the authors used in their paper. See Fig. 9 for a summary of the timings of our algorithm compared to the CCVT-based methods of [Xu et al. 2011; Chen et al. 2012] for the generation of blue noise sampling of a constant density field.

## 6 Future Work

We note that our numerical treatment is ripe for GPU implementations as each element (from power diagram construction to line search) is known to be parallelizable. The scalability of our approach should also make blue noise generation over non-flat surfaces and 3D volumes practical since our formulation and numerical approach generalizes to these cases without modification. Blue noise meshing is thus an obvious avenue to explore and evaluate for numerical benefits. On the theoretical side it would be interesting to seek a fully variational definition of blue noise that incorporates requirements A, B and C altogether. Generating anisotropic and multiclass sampling would also be desirable, as well as extending our regularity-breaking procedure to other CVT-based methods. Finally, the intriguing connection between HOT meshes [Mullen et al. 2011] and our definition of blue noise (which makes the Hodge-star for 0-forms not just diagonal, but constant) may deserve further exploration.

**Acknowledgements.** We wish to thank the authors of [Chen et al. 2012; Fattal 2011] for providing data for comparisons, and Christian Lessig for proof-reading. FdG, KB, and MD acknowledge the valuable support of NSF grants DGE-1147470 and CCF-1011944 throughout this project.



**Figure 10: Ramp.** Blue noise sampling of a quadratic density function with 1000 points. The percentages in each quarter indicate ink density in the image, and point density in the examples. Observe that our method returns the best matching of the reference percentages, while still presenting an even and unstructured distribution. Comparative data courtesy of the authors.

## References

- AURENHAMMER, F., HOFFMANN, F., AND ARONOV, B. 1998. Minkowski-type theorems and least-squares clustering. *Algorithmica* 20, 1, 61–76.
- BALZER, M., AND HECK, D. 2008. Capacity-constrained Voronoi diagrams in finite spaces. In *Int. Symp. on Voronoi Diag.*, 44–56.
- BALZER, M., DEUSSEN, O., AND LEWERENTZ, C. 2005. Voronoi treemaps for the visualization of software metrics. In *Symp. on Software Visualization*, ACM, 165–172.
- BALZER, M., SCHLÖMER, T., AND DEUSSEN, O. 2009. Capacity-constrained point distributions: A variant of Lloyd’s method. *ACM Trans. Graph. (SIGGRAPH)* 28, 3, 86:1–8.
- BALZER, M. 2009. Capacity-constrained Voronoi diagrams in continuous spaces. In *Int. Symp. on Voronoi Diagrams*, 79–88.
- BONNEEL, N., VAN DE PANNE, M., PARIS, S., AND HEIDRICH, W. 2011. Displacement interpolation using Lagrangian mass transport. *ACM Trans. Graph. (SIGGRAPH ASIA)* 30, 6.
- BOWERS, J., WANG, R., WEI, L.-Y., AND MALETZ, D. 2010. Parallel Poisson disk sampling with spectrum analysis on surfaces. *ACM Trans. Graph.* 29 (Dec.), 166:1–166:10.
- BRIDSON, R. 2007. Fast Poisson disk sampling in arbitrary dimensions. In *ACM SIGGRAPH sketches*.
- CGAL, 2010. Computational Geometry Algorithms Library (release 3.8). <http://www.cgal.org>.
- CHEN, Z., YUAN, Z., CHOI, Y.-K., LIU, L., AND WANG, W. 2012. Variational blue noise sampling. *IEEE Trans. Vis. Comput. Graphics* 18, 10, 1784–1796.
- COHEN, M. F., SHADE, J., HILLER, S., AND DEUSSEN, O. 2003. Wang tiles for image and texture generation. In *ACM SIGGRAPH*, 287–294.
- COOK, R. L. 1986. Stochastic sampling in computer graphics. *ACM Trans. Graph.* 5, 1, 51–72.
- CROW, F. C. 1977. The aliasing problem in computer-generated shaded images. *Commun. ACM* 20, 11, 799–805.
- DAVIS, T. A. 2011. Algorithm 915, SuiteSparseQR: Multifrontal multithreaded rank-revealing sparse QR factorization. *ACM Trans. Math. Softw.* 38 (Dec.), 8:1–8:22.
- DE GOES, F., COHEN-STEINER, D., ALLIEZ, P., AND DESBRUN, M. 2011. An optimal transport approach to robust reconstruction and simplification of 2d shapes. *Computer Graphics Forum* 30, 5, 1593–1602.
- DEUSSEN, O., HILLER, S., OVERVELD, C., AND STROTHOTTE, T. 2000. Floating points: A method for computing stipple drawings. *Computer Graphics Forum (EG’00)* 19, 3, 40–51.
- DIPPÉ, M. A. Z., AND WOLD, E. H. 1985. Antialiasing through stochastic sampling. In *ACM SIGGRAPH*, 69–78.
- DU, Q., FABER, V., AND GUNZBURGER, M. 1999. Centroidal Voronoi Tessellations: Applications and algorithms. *SIAM Rev.* 41 (Dec.), 637–676.
- DUNBAR, D., AND HUMPHREYS, G. 2006. A spatial data structure for fast Poisson-disk sample generation. *ACM Trans. Graph.* 25, 3 (July), 503–508.
- EBEIDA, M. S., DAVIDSON, A. A., PATNEY, A., KNUPP, P. M., MITCHELL, S. A., AND OWENS, J. D. 2011. Efficient maximal Poisson-disk sampling. *ACM Trans. Graph.* 30 (Aug.), 49:1–49:12.
- FATTAL, R. 2011. Blue-noise point sampling using kernel density model. *ACM Trans. Graph. (SIGGRAPH)* 30, 3, 48:1–48:12.
- FLOYD, R. W., AND STEINBERG, L. 1976. An adaptive algorithm for spatial grey scale. *Proc. Soc. Inf. Display* 17, 75–77.
- GAMITO, M. N., AND MADDOCK, S. C. 2009. Accurate multidimensional Poisson-disk sampling. *ACM Trans. Graph.* 29, 8:1–8:19.
- JONES, T. R. 2006. Efficient generation of Poisson-disk sampling patterns. *Journal of Graphics, GPU, & Game Tools* 11, 2, 27–36.
- KOPF, J., COHEN-OR, D., DEUSSEN, O., AND LISCHINSKI, D. 2006. Recursive Wang tiles for real-time blue noise. *ACM Trans. Graph.* 25, 3, 509–518.
- LAGAE, A., AND DUTRÉ, P. 2006. An Alternative for Wang Tiles: Colored Edges versus Colored Corners. *ACM Trans. Graph.*, 25, 4, 1442–1459.
- LAGAE, A., AND DUTRÉ, P. 2008. A comparison of methods for generating Poisson disk distributions. *Computer Graphics Forum* 27, 1, 114–129.
- LECOT, G., AND LÉVY, B. 2006. ARDECO: Automatic Region DEtection and CONversion. In *EG Symp. on Rendering*, 349–360.
- LEMIEUX, C. 2009. *Monte Carlo and Quasi Monte Carlo Sampling*. Springer.
- LI, H., NEHAB, D., WEI, L.-Y., SANDER, P., AND FU, C.-W. 2010. Fast capacity constrained Voronoi tessellation. In *Symp. on Interactive 3D Graphics & Games*, 13:1–13:4.
- LI, H., WEI, L.-Y., SANDER, P. V., AND FU, C.-W. 2010. Anisotropic blue noise sampling. *ACM Trans. Graph. (SIGGRAPH Asia)* 29 (Dec.), 167:1–167:12.
- LIU, Y., WANG, W., LÉVY, B., SUN, F., YAN, D., LU, L., AND YANG, C. 2009. On Centroidal Voronoi Tessellation - energy smoothness and fast computation. *ACM Trans. Graph.* 28, 4.
- LLOYD, S. 1982. Least squares quantization in PCM. *Information Theory, IEEE Transactions on* 28, 2 (Mar.), 129–137.

- LUCARINI, V. 2009. Symmetry-break in Voronoi tessellations. *Symmetry 1*, 1, 21–54.
- MCCOOL, M., AND FIUME, E. 1992. Hierarchical Poisson disk sampling distributions. In *Proc. Graphics Interface '92*, 94–105.
- MITCHELL, D. P. 1987. Generating antialiased images at low sampling densities. In *ACM SIGGRAPH*, 65–72.
- MULLEN, P., MEMARI, P., DE GOES, F., AND DESBRUN, M. 2011. HOT: Hodge Optimized Triangulations. *ACM Trans. Graph. (SIGGRAPH)* 30, 3.
- NIEDERREITER, H. 1992. *Random Number Generation and Quasi-Monte-Carlo Methods*. SIAM.
- NOCEDAL, J., AND WRIGHT, S. J. 1999. *Numerical optimization*. Springer Verlag.
- OSTROMOUKHOV, V., DONOHUE, C., AND JODOIN, P.-M. 2004. Fast hierarchical importance sampling with blue noise properties. *ACM Trans. Graph.* 23, 3, 488–495.
- OSTROMOUKHOV, V. 2007. Sampling with polyominoes. *ACM Trans. Graph.* 26, 3, 78:1–78:6.
- SCHLÖMER, T., AND DEUSSEN, O. 2011. Accurate spectral analysis of two-dimensional point sets. *Journal of Graphics, GPU, and Game Tools* 15, 3, 152–160.
- SCHLÖMER, T., HECK, D., AND DEUSSEN, O. 2011. Farthest-point optimized point sets with maximized minimum distance. In *Symp. on High Performance Graphics*, 135–142.
- SCHMALTZ, C., GWOSDEK, P., BRUHN, A., AND WEICKERT, J. 2010. Electrostatic halftoning. *Comput. Graph. Forum* 29, 8, 2313–2327.
- SECORD, A. 2002. Weighted Voronoi stippling. In *Symp. on Non-Photorealistic Animation and Rendering*, 37–43.
- ULICHNEY, R. 1987. *Digital Halftoning*. MIT Press.
- VILLANI, C. 2009. *Optimal Transport: Old and New*. Fundamental Principles of Mathematical Sciences, 338. Springer-Verlag.
- WEI, L.-Y., AND WANG, R. 2011. Differential domain analysis for non-uniform sampling. *ACM Trans. Graph.* 30 (Aug.), 50:1–50:10.
- WEI, L.-Y. 2008. Parallel Poisson disk sampling. *ACM Trans. Graph. (SIGGRAPH)* 27 (August), 20:1–20:9.
- WEI, L.-Y. 2010. Multi-class blue noise sampling. *ACM Trans. Graph. (SIGGRAPH)* 29 (July), 79:1–79:8.
- XIANG, Y., XIN, S.-Q., SUN, Q., AND HE, Y. 2011. Parallel and accurate Poisson disk sampling on arbitrary surfaces. In *SIGGRAPH Asia Sketches*, 18:1–18:2.
- XU, Y., LIU, L., GOTSMAN, C., AND GORTLER, S. J. 2011. Capacity-constrained Delaunay triangulation for point distributions. *Comput. Graph.* 35, 510–516.
- XU, Y., HU, R., GOTSMAN, C., AND LIU, L. 2012. Blue noise sampling of surfaces. *Comput. Graph.* 36, 232–240.
- YELLOTT, J. I. J. 1983. Spectral consequences of photoreceptor sampling in the rhesus retina. *Science* 221, 382–385.
- ZHOU, Y., HUANG, H., WEI, L.-Y., AND WANG, R. 2012. Point sampling with general noise spectrum. *ACM Trans. Graph. (SIGGRAPH)* 31, 4, 76:1–76:11.

## Appendix: Functional Properties and Derivatives

In this appendix, we provide closed-form expressions for the first and second derivatives of the functional  $\mathcal{F}$  defined in Eq. 2.

**Notation:** We denote by  $\mathbf{e}_{ij}$  the regular edge between two adjacent points  $\mathbf{x}_i$  and  $\mathbf{x}_j$ , and by  $\mathbf{e}_{ij}^*$  the dual edge separating the partition regions  $\mathcal{V}_i^w$  and  $\mathcal{V}_j^w$ . (Remember that  $\mathbf{x} \in \mathbf{e}_{ij}^*$  iff  $\|\mathbf{x} - \mathbf{x}_i\|^2 - w_i = \|\mathbf{x} - \mathbf{x}_j\|^2 - w_j$ .) We also refer to the average value of the field  $\rho$  over  $\mathbf{e}_{ij}^*$  as  $\bar{\rho}_{ij}$ , and to the one-ring of  $\mathbf{x}_i$  in the regular triangulation of  $(\mathbf{X}, W)$  as  $\Omega_i$ .

**Reynolds transport theorem:** The derivatives of  $\mathcal{F}$  are most directly found by Reynolds theorem, which states that the rate of change of the integral of a scalar function  $f$  within a volume  $V$  is equal to the volume integral of the change of  $f$ , plus the boundary integral of the rate at which  $f$  flows through the boundary  $\partial V$  of outward unit normal  $\mathbf{n}$ ; i.e., in terse notation:

$$\nabla \left( \int_V f(\mathbf{x}) dV \right) = \int_V \nabla f(\mathbf{x}) dV + \int_{\partial V} f(\mathbf{x}) (\nabla \mathbf{x} \cdot \mathbf{n}) dA.$$

**W.r.t. weights:** Since the regions  $\mathcal{V}_i^w$  partition the domain  $\mathcal{D}$ , the sum of all capacities is constant; hence,

$$\nabla_{w_i} m_i + \sum_{j \in \Omega_i} \nabla_{w_j} m_j = 0.$$

Moreover, Reynolds theorem applied to the capacities yields

$$\nabla_{w_i} m_j = -\frac{\bar{\rho}_{ij}}{2} \frac{|\mathbf{e}_{ij}^*|}{|\mathbf{e}_{ij}|}.$$

Next, by using both Reynolds theorem and the equality of power distances along dual edges, one obtains

$$\begin{cases} \nabla_{w_i} \mathcal{E}(\mathbf{X}, W) = \sum_{j \in \Omega_i} (w_j - w_i) (\nabla_{w_i} m_j) \\ \nabla_{w_i} \left[ \sum_j w_j (m_j - m) \right] = m_i - m + \sum_{j \in \Omega_i} (w_j - w_i) (\nabla_{w_i} m_j). \end{cases}$$

Therefore, the gradient simplifies to  $\nabla_{w_i} \mathcal{F}(\mathbf{X}, W) = m - m_i$ .

Combining the results above yields that the Hessian of  $\mathcal{F}$  with respect to weights is simply a negated weighted Laplacian operator:

$$\nabla_{w_i}^2 \mathcal{F}(\mathbf{X}, W) = -\Delta^{w, \rho} \quad \text{with } [\Delta^{w, \rho}]_{ij} = -\frac{\bar{\rho}_{ij}}{2} \frac{|\mathbf{e}_{ij}^*|}{|\mathbf{e}_{ij}|}$$

For fixed points,  $\mathcal{F}$  is thus a concave function in weights and there is a unique solution  $W_{\text{opt}}$  for any prescribed capacity constraints.

**W.r.t. position:** We first note that  $\nabla_{\mathbf{x}_i} m_i + \sum_{j \in \Omega(i)} \nabla_{\mathbf{x}_i} m_j = 0$  as in the weight case. Using the definition of the weighted barycenter  $\mathbf{b}_i$  (Eq. 3), Reynolds theorem then yields

$$\begin{cases} \nabla_{\mathbf{x}_i} \mathcal{E}(\mathbf{X}, W) = 2 m_i (\mathbf{x}_i - \mathbf{b}_i) + \sum_{j \in \Omega_i} (w_j - w_i) (\nabla_{\mathbf{x}_i} m_j) \\ \nabla_{\mathbf{x}_i} \left[ \sum_j w_j (m_j - m) \right] = \sum_{j \in \Omega_i} (w_j - w_i) (\nabla_{\mathbf{x}_i} m_j). \end{cases}$$

Therefore:  $\nabla_{\mathbf{x}_i} \mathcal{F}(\mathbf{X}, W) = 2m_i(\mathbf{x}_i - \mathbf{b}_i)$ .

**Equivalence of Optimizations:** The constrained minimization with Lagrangian multipliers (Eq. 1) is equivalent to extremizing the functional  $\mathcal{F}$  (Eq. 2). Indeed, observe that any solution of the Lagrangian formulation is a stationary point of the functional  $\mathcal{F}$ , since we just derived that a null gradient implies that  $m_i = m$  (constraints are met) and  $\mathbf{x}_i = \mathbf{b}_i$  (centroidal power diagram). Another way to understand this equivalence is to observe that the gradient with respect to weights of the Lagrangian formulation is  $\Delta^{w, \rho} (W + \Lambda)$ ; hence, extremization induces that  $W = -\Lambda + \text{constant}$ , and the Lagrange multipliers can be directly replaced by the (negated) weights.