Sander Scholtus

Weighted Distance Transformations with Integer Neighbourhoods

Master Thesis
(Doctoraalscriptie)

Supervised by Prof. Dr. R. Tijdeman
Exam date: August 21, 2006

Mathematisch Instituut, Universiteit Leiden

Table of contents

Introduction

The problem of measuring distances in digital pictures efficiently has been studied by many authors from a variety of research angles. Weighted distance transformations as a solution originally date back to the late 1960's (cf. [14]–[16]), but did not come under intensive study until the mid 1980's, when they were introduced in a more general form by Borgefors (cf. [2]–[5]). The advantage of these distance transformations is that they are efficient and easy to implement. The main disadvantage is their lack of accuracy.

Over the past twenty years different approaches have been used to find the best weighted distance transformations. (We provide a brief overview in Section 1.4.) Recently Hajdu, Hajdu and Tijdeman [11] have studied the problem from a purely theoretical point of view. In this thesis we use their results to construct uniform classes of weighted distance transformations, with guaranteed bounds on the inaccuracy. It so happens that every good weighted distance transformation that has been suggested previously falls into one of these classes.

A word on the structure of this thesis. Chapter 1 contains a description of the theory of weighted distance transformations. Concepts, notation and terminology that will be used in the rest of the thesis are introduced. Much of this is non-standard, as most previous authors have used their own notation. When possible we have adopted the terminology of Hajdu, Hajdu and Tijdeman. Also a measure of quality called the *maximum relative error* is defined.

In Chapter 2 a scheme to calculate (a bound on) this maximum relative error is established.

Five classes of weighted distance transformations are defined in Chapters 3 and 4, and it is shown that the results of Chapter 2 are valid for these classes.

Finally Chapter 5 contains tables of the best weighted distance transformations from each class.

Chapter 1

The theory of weighted distance transformations

1.1 Distance transformations in two dimensions

In applications of pattern recognition and image analysis, it is often necessary to measure distances between pixels of digital pictures. Examples include matching (see [4] for a working algorithm), skeletonising (see [7], [9], [14] and [18]) and segmentation (see [9]). In this thesis we shall study the theory of distance measuring and we will not deal with such applications. The interested reader is referred to the publications mentioned above. An extensive bibliography of applications, particularly from the medical world, can be found in [9].

Before a picture is analysed by a computer, it is converted into a digital *binary picture*. This is done by placing a grid of pixels on top of the original picture, and assigning one of two possible values to every pixel in the grid (hence the name binary picture). The grid is thus partitioned into *feature* and *non-feature pixels*. Non-feature pixels are also referred to as *background pixels*. (See Figure 1.1 for an illustration of this process.)

A digital picture is a discretisation of the original, continuous picture. Taking the size of a pixel[1] as a unit of length, we can think of the pixel-grid as a subset of $\mathbb{Z}^2$. The digital binary picture is then described by assigning a binary value to every point in that subset.

The next step is to compute for every pixel the distance to the nearest feature pixel (which is defined to be 0 for the feature pixels themselves). This information is stored in a new digital object, called the *distance map* (for obvious reasons). The operation converting a binary picture to a distance map is called a *distance transformation*. In order to get it, we have to be

[1]We will assume throughout that the pixels are square. While most authors make this assumption, in some applications it is natural to work with non-square (rectangular) pixels. See e.g. [7] and [8] for a study of distance transformations on rectangular grids.

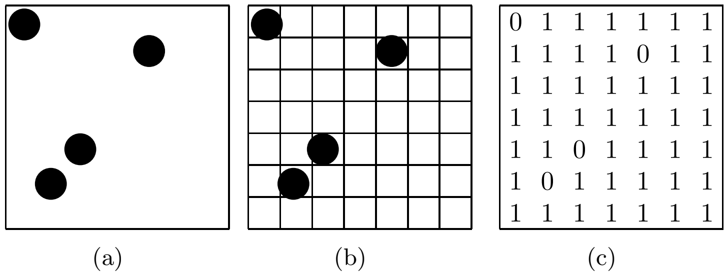

Figure 1.1: An illustration of how a picture is digitised. The original picture (a) is covered by a pixel-grid, as seen in (b). The corresponding digital binary picture (c) is now found by assigning (in this case) the value 0 to the feature pixels and 1 to the background pixels.

able to compute the distance between points of $\mathbb{Z}^2$. The relevant distance is usually the Euclidean distance, d_{E}, given by

$$d_{\mathrm{E}}(\vec{u},\vec{v}) = \sqrt{(u_x - v_x)^2 + (u_y - v_y)^2}, \tag{1.1}$$

where $\vec{u} = (u_x, u_y)$ and $\vec{v} = (v_x, v_y)$ are vectors in a two-dimensional space.

Obviously, we could use formula (1.1) directly for every pair of points in our subset of $\mathbb{Z}^2$, and apply this to our binary picture to get a distance transformation, but in practice this is simply too much work, since it involves the evaluation of many square roots in real arithmetic. In order to keep the computational effort low, it is desirable to restrict the calculations to integers and also to restrict the amount of calculations needed.

An easy solution for the first problem would be to make a simple modification of the Euclidean distance, in one of the following ways:

$$\begin{aligned}
(d_{\mathrm{E}}(\vec{u},\vec{v}))^2 &= (u_x - v_x)^2 + (u_y - v_y)^2, \\
\langle d_{\mathrm{E}}(\vec{u},\vec{v}) \rangle &= \langle \sqrt{(u_x - v_x)^2 + (u_y - v_y)^2} \rangle, \\
\lfloor d_{\mathrm{E}}(\vec{u},\vec{v}) \rfloor &= \lfloor \sqrt{(u_x - v_x)^2 + (u_y - v_y)^2} \rfloor.
\end{aligned}$$

(Here, $\langle x \rangle$ denotes x rounded off to the nearest integer, with the usual convention of rounding up odd multiples of $\frac{1}{2}$, and $\lfloor x \rfloor$ is the largest integer smaller than or equal to x. For future reference, $\lceil x \rceil$ denotes the smallest integer larger than or equal to x.)

All three choices yield only integer values when $\vec{u}, \vec{v} \in \mathbb{Z}^2$. However, neither one of these is a *metric*. We recall that a function $d : A \times A \to \mathbb{R}$ is called a metric if it satisfies the following three conditions for all $\vec{u}, \vec{v}, \vec{w} \in A$:

$$\begin{gathered}
d(\vec{u},\vec{v}) = 0 \text{ if and only if } \vec{u} = \vec{v} \text{ (positive definiteness)}, \\
d(\vec{u},\vec{v}) = d(\vec{v},\vec{u}) \text{ (symmetry)}, \\
d(\vec{u},\vec{w}) \leq d(\vec{u},\vec{v}) + d(\vec{v},\vec{w}) \text{ (triangle inequality)}.
\end{gathered} \tag{1.2}$$

Also, we call $w : A \to \mathbb{R}$ a *length function* if it satisfies, for all $\vec{u}, \vec{v} \in A$:

$$\begin{aligned} w(\vec{u}) = 0 \text{ if and only if } \vec{u} = \vec{0} \text{ (positive definiteness)},\\ w(-\vec{u}) = w(\vec{u}) \text{ (symmetry)},\\ w(\vec{u}+\vec{v}) \le w(\vec{u}) + w(\vec{v}) \text{ (triangle inequality)}. \end{aligned} \tag{1.3}$$

(For clarity, we use $\vec{0}$ to denote the zero vector.) We remark that if d is a metric, the projection $w(\vec{v}) := d(\vec{0}, \vec{v})$ is a length function, and that if w is a length function, $d(\vec{u}, \vec{v}) := w(\vec{u} - \vec{v})$ defines a metric.

The modified Euclidean "distances" given above are not metrics, since they do not always satisfy the triangle inequality. (For $\langle d_{\mathrm{E}}(\vec{u}, \vec{v})\rangle$ and $\lfloor d_{\mathrm{E}}(\vec{u}, \vec{v})\rfloor$, counter-examples are provided in [16], one of the first articles on distance transformations.)

Another serious drawback of using formula (1.1) directly, is that in practice the number of pairs of pixels in the binary picture to be considered is so large, that it is too costly to perform such a *global operation*. In this thesis, we shall deal with a type of solution which is computationally cheaper, and approximates the Euclidean distance using only *local operations*, i.e. working with a small neighbourhood of points. This leads to a class of distance transformations called *weighted distance transformations*. (The name *chamfer distances* is also commonly used.) Although the resulting distance map is inaccurate in comparison with the true Euclidean distance, the approximation can be made arbitrarily close, at the cost of increasing computational work. However, a fair approximation often suffices in practice, as there are flaws and uncertainties in the original picture.

We remark that there exist various ways of working with the exact[2] Euclidean distance in $\mathbb{Z}^2$ in integer arithmetic while using only local operations. (See e.g. [10] and [12].) The algorithms involved are more complex than those of weighted distance transformations, and have the disadvantage that they are less efficient.[3]

1.2 Weighted distance transformations

The idea behind weighted distance transformations is to prescribe lengths of vectors from $\mathbb{Z}^2$ in some small neighbourhood of the origin, and to use these

[2]Actually, most implementations of these *Euclidean distance transforms* are not guaranteed to be flawless. If errors do occur, however, they are exceptional and their size is bounded by a fixed constant (that does not depend on the size of the picture), as opposed to the errors we find using weighted distance transformations.

[3]In [12] Leymarie and Levine claim that Euclidean distance transforms can be implemented just as efficiently as weighted distance transformations. They admit that these (careful) implementations require three times as much computer memory, but argue that the additional information we can obtain thus is worth the extra cost. Of course this depends on the application at hand. Weighted distance transformations still appear to be a good solution for simple tasks.

2	**1**	**2**
1	∞	**1**
2	**1**	**2**

(a)

1	**1**	**1**
1	∞	**1**
1	**1**	**1**

(b)

4	**3**	**4**
3	∞	**3**
4	**3**	**4**

(c)

14	**11**	10	**11**	14
11	**7**	**5**	**7**	**11**
10	**5**	∞	**5**	10
11	**7**	**5**	**7**	**11**
14	**11**	10	**11**	14

(d)

Figure 1.2: Examples of commonly used neighbourhoods with $p = 1$ (a–c) and $p = 2$ (d). The values of the visible points are printed in boldface.

local vectors as stepping stones to the other points of $\mathbb{Z}^2$. Thus, the global distance transformation is found by propagating a locally defined distance transformation.

Let $p \geq 1$ be an integer. The *mask* of size $(2p+1) \times (2p+1)$ is defined as $M_p := \{(i,j) \in \mathbb{Z}^2 : |i| \leq p, |j| \leq p\}$. We also define $M_p^* := M_p \setminus \{(0,0)\}$.

A *neighbourhood* w is defined on M_p by attributing a positive value (a 'weight') to every vector of M_p^*. (The origin itself gets the value infinity, if we want to avoid its use as a valid step.) It is very natural to demand that the neighbourhood be symmetric, i.e. that $w(i,j) = w(i,-j) = w(-i,j) = w(-i,-j)$ for every $(i,j) \in M_p$, and also that $w(i,j) = w(j,i)$ for such (i,j). Under these assumptions, a neighbourhood is completely determined by its values on the first octant of M_p, where $0 \leq j \leq i \leq p$.

Some commonly used distance transformations are given by the neighbourhoods displayed in Figure 1.2. The first two of these are often referred to as the *city block* and *chessboard* distance transformations, respectively. The third and fourth are due to Borgefors (see [2] and [3]). Notice that in the last neighbourhood, the values 10 and 14 are actually redundant, since they are generated by the other values. In general, to define a neighbourhood w it is sufficient to prescribe the lengths of the so-called *visible points* of M_p, which are the pairs (i,j) such that i and j are coprime. All other elements of w can be inferred from these.

The notation of Figure 1.2 is slightly misleading, because the elements of w are not integers, but integer multiples of a common real number $\frac{1}{s}$:

$$w(i,j) = \frac{N(i,j)}{s} \qquad \left((i,j) \in M_p^*\right). \tag{1.4}$$

The values displayed in Figure 1.2 are the numerators $N(i,j)$. In these examples the *scaling factor* s is always equal to $N(1,0)$ – so the vector $(1,0)$ gets a weight of 1 –, but in general it may be any real number. Of course, when we interpret the value $w(i,j)$ as an approximation of the true Euclidean length $\sqrt{i^2+j^2}$, the scaling factor has to be divided out before we make the comparison: e.g. in the examples given in Figure 1.2, the Euclidean length $\sqrt{2}$ is approximated by (in order of appearance) 2, 1, $\frac{4}{3}$ and $\frac{7}{5}$.

This short-hand notation – which will be used throughout this thesis – is also useful in practice. Since every element of w gets the same scaling factor, we can defer the division until the end of the calculation. If the numerators $N(i,j)$ are all positive integers, this enables us to work with the distance transformation using integers only, thereby reducing the computational work-load. No real arithmetic is needed until the final step. In that case we call N an *integer neighbourhood*.

Let $\vec{u}_1, \ldots, \vec{u}_n$ be vectors in M_p^*. Concatenation of these vectors yields a path $P = [\vec{u}_1, \ldots, \vec{u}_n]$ in $\mathbb{Z}^2$. For an integer neighbourhood N defined on M_p, the length of this path is defined as the sum of all the associated values:

$$\ell(P) = \frac{1}{s} \sum_{i=1}^{n} N(\vec{u}_i). \tag{1.5}$$

The empty path gets length zero: $\ell(\emptyset) = 0$.

The function N now induces a distance d on the whole of $\mathbb{Z}^2$, by taking $d(\vec{u}, \vec{v})$ as the minimal length over all possible paths from $\vec{u}$ to $\vec{v}$ composed solely of steps from M_p. In particular, $w = \frac{N}{s}$ is extended to a length function w on the whole of $\mathbb{Z}^2$, by taking $w(\vec{v})$ as the minimal length over all possible paths from the origin to $\vec{v}$ composed of steps from M_p.

A proof of the metricity of d, under the conditions of symmetry we imposed on w (and thus also on N), was given by Yamashita and Ibaraki in [24]. (A similar proof was given by Verwer in [21].) We restate their proof here using a notation that is consistent with the rest of this thesis.

Lemma 1.1 (Yamashita and Ibaraki, Verwer) *The distance function d induced by any neighbourhood w defined on any mask M_p is a metric*[4].

Proof. We check that the three properties of a metric are satisfied. Positive definiteness is trivial and symmetry is guaranteed by the conditions of symmetry we imposed on w. It remains to show that d satisfies the triangle inequality. Let $\vec{u}_1, \vec{u}_2, \vec{u}_3 \in \mathbb{Z}^2$ and let P and Q be shortest paths from $\vec{u}_1$ to $\vec{u}_2$ and from $\vec{u}_2$ to $\vec{u}_3$, respectively. We construct a path R from $\vec{u}_1$ to $\vec{u}_3$ by concatenating P and Q. Then, by definition of d, we have

$$d(\vec{u}_1, \vec{u}_3) \leq \ell(R) = \ell(P) + \ell(Q) = d(\vec{u}_1, \vec{u}_2) + d(\vec{u}_2, \vec{u}_3).$$

So the triangle inequality is satisfied and d is a metric. □

As an example, Figure 1.3 shows the values near $(0,0)$ of the length function induced by the integer neighbourhood with $p = 2$ given in Figure 1.2(d). Again, to compare this with the true Euclidean length, we first have to divide by (in this case) 5. So for instance, $\sqrt{4^2 + 2^2} = \sqrt{20} \approx 4.47$ is approximated by $\frac{22}{5} = 4.4$.

[4] This lemma shows that w induces a metric *regardless* of the values we attribute to M_p (provided they are positive). Note however that if we had chosen these values at random, the resulting metric would not even remotely resemble the Euclidean one.

35	32	29	27	26	25	26	27	29	32	35
32	28	25	22	21	20	21	22	25	28	32
29	25	21	18	16	15	16	18	21	25	29
27	22	18	14	11	10	11	14	18	22	27
26	21	16	11	7	5	7	11	16	21	26
25	20	15	10	5	0	5	10	15	20	25
26	21	16	11	7	5	7	11	16	21	26
27	22	18	14	11	10	11	14	18	22	27
29	25	21	18	16	15	16	18	21	25	29
32	28	25	22	21	20	21	22	25	28	32
35	32	29	27	26	25	26	27	29	32	35

Figure 1.3: Example of a length function. The original neighbourhood is enclosed by the box.

1.3 Computing distance maps

For a given integer neighbourhood N there exist two ways to construct a distance map $\mathcal{D}$ from a digital binary picture $\mathcal{B}$: it can be done by either a *parallel* or a *sequential* algorithm. For the M_1-mask these algorithms were published by Rosenfeld and Pfaltz in [16] (the parallel case) and [15] (the sequential case). The extension to larger masks is straightforward. A good description of both algorithms for general masks is given by Borgefors in [3].

In both cases we need to calculate for each pixel in the binary picture $\mathcal{B}$ the length of the shortest path to the nearest feature pixel, where "length" is defined by equation (1.5). Please note that in the description of the algorithms given below we ignore the scaling factor s, which should be divided out after the (integer-valued) distance map has been computed.

The parallel algorithm constructs a sequence of distance maps $\mathcal{D}_0$, $\mathcal{D}_1$, $\mathcal{D}_2$, ..., in which the local distance transformation described by N is propagated further and further across the pixel-grid. The first map $\mathcal{D}_0$ is obtained by setting

$$\mathcal{D}_0(i,j) := \begin{cases} 0 & \text{if } \mathcal{B}(i,j) = 0 \\ \infty & \text{if } \mathcal{B}(i,j) = 1 \end{cases} \tag{1.6}$$

and subsequent distance maps are found by computing for each pixel (i,j)

$$\mathcal{D}_n(i,j) := \min\left(\mathcal{D}_{n-1}(i,j), \min_{(k,l)\in M_p^*} \{\mathcal{D}_{n-1}(i+k, j+l) + N(k,l)\}\right).$$

The algorithm stops when the current iteration yields no changes, i.e. when $\mathcal{D}_n = \mathcal{D}_{n-1}$. Figure 1.4 shows an illustration of the parallel algorithm.

In the sequential algorithm the mask M_p is split in two parts, called the *forward* and *backward* masks. If we count off all the points in M_p, starting at $(-p,p)$ – i.e. the upper left-hand corner – and going from left to right and

0	∞	∞	∞	∞	∞	∞
∞	∞	∞	∞	0	∞	∞
∞	∞	∞	∞	∞	∞	∞
∞	∞	∞	∞	∞	∞	∞
∞	∞	0	∞	∞	∞	∞
∞	0	∞	∞	∞	∞	∞
∞	∞	∞	∞	∞	∞	∞

(a)

0	3	∞	4	3	4	∞
3	4	∞	3	0	3	∞
∞	∞	∞	4	3	4	∞
∞	4	3	4	∞	∞	∞
4	3	0	3	∞	∞	∞
3	0	3	4	∞	∞	∞
4	3	4	∞	∞	∞	∞

(b)

0	3	6	4	3	4	7
3	4	6	3	0	3	6
6	7	6	4	3	4	7
7	4	3	4	6	7	8
4	3	0	3	6	∞	∞
3	0	3	4	7	∞	∞
4	3	4	7	8	∞	∞

(c)

0	3	6	4	3	4	7
3	4	6	3	0	3	6
6	7	6	4	3	4	7
7	4	3	4	6	7	8
4	3	0	3	6	9	11
3	0	3	4	7	10	∞
4	3	4	7	8	11	∞

(d)

0	3	6	4	3	4	7
3	4	6	3	0	3	6
6	7	6	4	3	4	7
7	4	3	4	6	7	8
4	3	0	3	6	9	11
3	0	3	4	7	10	13
4	3	4	7	8	11	14

(e)

Figure 1.4: An illustration of the parallel algorithm. The distance transformation of Figure 1.2(c) is applied to the binary picture of Figure 1.1. Figures (a)–(e) show $\mathcal{D}_0$, $\mathcal{D}_1$, $\mathcal{D}_2$, $\mathcal{D}_3$ and finally $\mathcal{D}_4$, which is the resulting distance map.

from top to bottom, the forward mask consists of all points up to – but *not* including – the origin, and the backward mask consists of all points beyond the origin:

$$M_p^{\text{forward}} := \{(-p,p),(-p+1,p),\ldots,(p,p),(-p,p-1),\ldots,(-1,0)\},$$
$$M_p^{\text{backward}} := \{(1,0),(2,0),\ldots,(p,0),(-p,-1),\ldots,(p,-p)\}.$$

Note that the origin does not fall into either half-mask.

The sequential algorithm starts with the map $\mathcal{D}_0$ defined by (1.6). Here only two iterations are needed to find the distance map. In the first iteration a map $\mathcal{D}_1$ is constructed by applying the forward mask to each pixel of $\mathcal{D}_0$, going from left to right and from top to bottom:

$$\mathcal{D}_1(i,j) := \min\left(\mathcal{D}_0(i,j), \min_{(k,l)\in M_p^{\text{forward}}} \{\mathcal{D}_1(i+k,j+l)+N(k,l)\}\right).$$

In the second iteration the backward mask is applied to each pixel of $\mathcal{D}_1$, this time starting at the lower right-hand corner of the picture and going from right to left and from bottom to top:

$$\mathcal{D}_2(i,j) := \min\left(\mathcal{D}_1(i,j), \min_{(k,l)\in M_p^{\text{backward}}} \{\mathcal{D}_2(i+k,j+l)+N(k,l)\}\right).$$

0	∞	∞	∞	∞	∞	∞
∞	∞	∞	∞	0	∞	∞
∞	∞	∞	∞	∞	∞	∞
∞	∞	∞	∞	∞	∞	∞
∞	∞	0	∞	∞	∞	∞
∞	0	∞	∞	∞	∞	∞
∞	∞	∞	∞	∞	∞	∞

(a)

0	3	6	9	12	15	18
3	4	7	10	0	3	6
6	7	8	4	3	4	7
9	10	8	7	6	7	8
12	12	0	3	6	9	11
15	0	3	4	7	10	13
4	3	4	7	8	11	14

(b)

0	3	6	4	3	4	7
3	4	6	3	0	3	6
6	7	6	4	3	4	7
7	4	3	4	6	7	8
4	3	0	3	6	9	11
3	0	3	4	7	10	13
4	3	4	7	8	11	14

(c)

Figure 1.5: An illustration of the sequential algorithm. The distance transformation of Figure 1.2(c) is applied to the binary picture of Figure 1.1. Figures (a)–(c) show $\mathcal{D}_0$, $\mathcal{D}_1$ and $\mathcal{D}_2$, which is the resulting distance map.

The resulting map $\mathcal{D}_2$ is the desired distance map. Figure 1.5 illustrates the sequential algorithm.

The number of iterations for the parallel algorithm depends on the size of the picture, whereas the sequential algorithm always needs only two iterations. Clearly on a single machine the sequential algorithm is usually much faster. However, the parallel algorithm has the advantage that the work per iteration can be split up over several machines, which is impossible in the sequential case.

It was established in [15] that the two algorithms are equivalent. Indeed we can see in Figures 1.4 and 1.5 that the computed distance maps for our example are identical. However, Thiel demonstrated in [18] that the equivalence is not true for all neighbourhoods: unless certain (mild) restrictions on the neighbourhood values are satisfied, the sequential algorithm may produce a map where some values differ from the true distance map.[5]

1.4 A brief history of optimal weighted distance transformations

Following the terminology of [11], we will consider three different classes of neighbourhoods. The superscript B is reserved for the case where we demand that $w(\vec{v}) = \|\vec{v}\|$ for all $\vec{v} = (v_x, 0)$, where $\|.\|$ denotes Euclidean length: $\|\vec{v}\| = d_{\mathrm{E}}(\vec{0}, \vec{v})$. That is to say, all points lying on the horizontal axis (and by virtue of symmetry also the vertical axis) get a weight equal to the exact Euclidean distance from the origin.

[5]Thiel claims that his restrictions are necessary to guarantee that the induced distance function of a neighbourhood is a metric, but this would contradict Lemma 1.1. In fact he merely shows that the distance function found by the sequential algorithm is not a metric in these exceptional cases. The parallel algorithm *does* produce the correct distance function, which is still a metric.

The second class is indicated by D, and consists of neighbourhoods that always overestimate the Euclidean distance: $d(\vec{u},\vec{v}) \geq d_{\mathrm{E}}(\vec{u},\vec{v})$ for all $\vec{u},\vec{v} \in \mathbb{Z}^2$. Distance transformations of this type are used in applications where it is vital that distances are not underestimated, such as collision avoidance in robotics (see [21]).

We use a superscript C for the most general problem, where no a priori assumption is made at all on the elements of the neighbourhood.

To compare the approximations to the Euclidean distance of different weighted distance transformations, we introduce a measure of quality called the *maximum relative error*. For a neighbourhood with induced length function w, this error is defined by

$$e = \limsup_{\|\vec{v}\|\to\infty} \left| \frac{w(\vec{v})}{\|\vec{v}\|} - 1 \right|. \tag{1.7}$$

For instance, the distance transformation in Figure 1.3 has a maximum relative error of approximately 0.0198. It is not trivial to calculate e; this calculation will be the subject of the next chapter.

Maybe contrary to intuition, the maximum relative error is not minimised (in the B- and C-cases) by setting each neighbourhood value equal to the true Euclidean length of the corresponding vector. (This distance transformation was suggested in [14].) It is therefore a non-trivial problem to construct optimal weighted distance transformations for a given mask-size p.

This topic was pioneered by Borgefors in the 1980's, in [2] and [3] for the B-case and in [5] for the C-case, but using a different optimisation criterion, namely to minimise the *maximum absolute error*:

$$e_{\mathrm{abs}} = \sup_{j\leq M} \left| w\left((M,j)\right) - \sqrt{M^2+j^2} \right|, \tag{1.8}$$

for some large $M \in \mathbb{Z}_{>0}$. Borgefors treated mask-sizes $p=1$, $p=2$ and $p=3$, and also gave good choices of integer neighbourhood for these mask-sizes.

In his 1991 paper [22] Verwer computed the optimal w for the C-case for all values of p, with respect to the maximum relative error. Besides he gave several integer neighbourhoods for $p=1$ and $p=2$.

In his 1994 PhD thesis [18] Thiel presented numerous examples of integer neighbourhoods for $p=2$, $p=3$ and $p=6$. Instead of first deriving optimal neighbourhoods and then using scaling factors to get integer approximations (which was the approach used by Borgefors and Verwer), Thiel constructed integer neighbourhoods directly and found the best choices by trying all scaling factors up to 256. His method works for every p, but is unpractical for large masks.

Coquin and Bolon extended the theory to pixels on a rectangular lattice instead of a square one in [8]. All their integer neighbourhoods refer to the square case, with $p = 1$, $p = 2$ and $p = 3$. Their method is an adaptation of Borgefors' method, this time with optimisation with respect to the maximum relative error. They treated both the B- and C-cases.

Butt and Maragos used a geometric approach in their 1998 paper [6] to find optimal neighbourhoods for the C-case, with integer examples for $p = 1$ and $p = 2$. Instead of the maximum relative error they used the following, closely related measure:

$$e_{\mathrm{BM}} = \limsup_{\|\vec{v}\| \to \infty} \left| 1 - \frac{\|\vec{v}\|}{w(\vec{v})} \right| . \tag{1.9}$$

While this value obviously differs from the maximum relative error in general, it is interesting to note that Butt and Maragos' argument results in the same *optimal* value for the error as the one found by Verwer (equation (1.12) below).[6] We give a proof of this fact in Appendix A.1.

In unpublished work [11] Hajdu, Hajdu and Tijdeman have determined the optimal values of the maximum relative error for all p, for all three cases. Below we will summarise these results, which are consistent with Verwer's and Coquin and Bolon's.

For $p \geq 1$, the following optimal values of the maximum relative error were derived in [11]:

$$e_p^B = \frac{p^2 + 2 - p\sqrt{p^2+1} - 2\sqrt{p^2 + 1 - p\sqrt{p^2+1}}}{p^2}, \tag{1.10}$$

$$e_p^D = \sqrt{\left(\sqrt{p^2+1} - p\right)^2 + 1} - 1, \tag{1.11}$$

$$e_p^C = \frac{e_p^D}{2 + e_p^D} = \frac{\sqrt{2p^2 + 2 - 2p\sqrt{p^2+1}} - 1}{\sqrt{2p^2 + 2 - 2p\sqrt{p^2+1}} + 1}. \tag{1.12}$$

Table 1.1 shows rounded values of e_p^B, e_p^D and e_p^C for small values of p. As may be expected, the unrestricted C-case yields smaller optima than the B-case, which in turn surpasses the D-case, where we set the most severe restriction.

Neighbourhoods that achieve the optimal maximum relative error are also given in [11]. In the B-case, the following neighbourhood has maximum relative error e_p^B:

$$w_p^B(i,j) = \begin{cases} |i| & \text{if } 1 \leq |i| \leq p,\ j = 0 \\ |j| & \text{if } 1 \leq |j| \leq p,\ i = 0 \\ \left(1 - e_p^B\right)\sqrt{i^2 + j^2} & \text{for all other vectors in } M_p^* \end{cases} \tag{1.13}$$

[6]Butt and Maragos seem to have overlooked this fact themselves, as they write that their optimal error values "are different from the values obtained by Verwer".

Table 1.1: Approximate values of the optimal maximum relative error in the B-, D- and C-cases, for $1 \leq p \leq 10$.

p	e_p^B	e_p^D	e_p^C
1	0.05505271	0.08239220	0.03956613
2	0.01869475	0.02748630	0.01355683
3	0.00893928	0.01308146	0.00649823
4	0.00516800	0.00754900	0.00376031
5	0.00335091	0.00489047	0.00243927
6	0.00234378	0.00341897	0.00170657
7	0.00172949	0.00252214	0.00125948
8	0.00132791	0.00193614	0.00096713
9	0.00105127	0.00153258	0.00076570
10	0.00085272	0.00124302	0.00062112

The D-case is minimised by the true Euclidean distance:

$$w_p^D(i,j) = \sqrt{i^2 + j^2} \qquad \left((i,j) \in M_p^*\right) \tag{1.14}$$

has maximum relative error e_p^D. Finally, the C-case is optimised similarly to the B-case; the following neighbourhood has maximum relative error e_p^C:

$$w_p^C(i,j) = \left(1 - e_p^C\right) \sqrt{i^2 + j^2} \qquad \left((i,j) \in M_p^*\right). \tag{1.15}$$

These neighbourhoods can not be written in terms of integer neighbourhoods and therefore have only theoretical value. One of the main purposes of this thesis is to provide good approximating integer neighbourhoods to w_p^B, w_p^D and w_p^C.

Chapter 2

Calculating the maximum relative error

In this chapter we will derive an easy-to-use expression for the maximum relative error of an integer neighbourhood satisfying mild restrictions.

2.1 The case $p = 1$

The case $p = 1$ differs from the general case at some points, and will therefore be treated separately. (This decision is also prompted by the simplicity of the treatment for $p = 1$, compared to the general case.)

Under the conditions of symmetry stated in Section 1.2, the general form of an integer neighbourhood N on the 3×3-mask M_1 is:

$$N(i,j) = \begin{cases} n_0 & \text{if } |i| + |j| = 1 \\ n_1 & \text{if } |i| + |j| = 2 \end{cases} \tag{2.1}$$

where n_0 and n_1 are positive integers. This neighbourhood is displayed in Figure 2.1. A scaling factor s is associated to N.

We recall that any choice of n_0 and n_1 will lead to a metric on $\mathbb{Z}^2$ (cf. Lemma 1.1). For convenience we shall impose the following mild constraint on the elements of N:

$$n_0 \leq n_1 \leq 2n_0. \tag{2.2}$$

$$N: \quad \begin{matrix} n_1 & n_0 & n_1 \\ n_0 & \infty & n_0 \\ n_1 & n_0 & n_1 \end{matrix}$$

Figure 2.1: General form of an integer neighbourhood on M_1.

If (2.2) were not satisfied, the resulting metric would behave very much unlike the Euclidean metric: when $n_1 > 2n_0$, for instance, the diagonal elements of N are not used at all, since it is cheaper to use a combination of a horizontal and a vertical step instead.

The maximum relative error of N was defined by (1.7) and can be written as follows:

$$e = \max\left\{1 - \liminf_{\|\vec{v}\|\to\infty} \frac{w(\vec{v})}{\|\vec{v}\|},\ \limsup_{\|\vec{v}\|\to\infty} \frac{w(\vec{v})}{\|\vec{v}\|} - 1\right\} =: \max\left\{e_{\min},\ e_{\max}\right\}.$$

By virtue of symmetry, it suffices to examine the behaviour of the distance function on the first octant of $\mathbb{Z}^2$. Condition (2.2) clearly implies that a path of shortest length from the origin to (m,k), with $0 \le k \le m$, consists of k steps $(1,1)$ and $m-k$ steps $(1,0)$. The induced length function w is thus given by

$$w(m,k) = \frac{1}{s}\left\{kn_1 + (m-k)n_0\right\} \qquad (0 \le k \le m).$$

Dividing this expression by the true Euclidean length of (m,k), we find:

$$\frac{w(m,k)}{\sqrt{m^2+k^2}} = \frac{1}{s}\frac{k(n_1-n_0)+mn_0}{m\sqrt{1+(k/m)^2}}.$$

This is written as a univariate function by introducing a new variable $t = \frac{k}{m}$; we call this function h:

$$h(t) := \frac{1}{s}\frac{(n_1-n_0)t+n_0}{\sqrt{1+t^2}} \qquad (0 \le t \le 1).$$

It is obvious that

$$\limsup_{m,k\ge 0} \frac{w(m,k)}{\sqrt{m^2+k^2}} = \max_{0\le t\le 1} h(t), \qquad \liminf_{m,k\ge 0} \frac{w(m,k)}{\sqrt{m^2+k^2}} = \min_{0\le t\le 1} h(t),$$

and the problem of finding $e_{\min}$ and $e_{\max}$ has been reduced to an ordinary optimisation problem.

We find the maximum and minimum of h through basic calculus. First, we take the derivative:

$$h'(t) = \frac{1}{s}\frac{n_1-n_0-n_0t}{(1+t^2)^{3/2}}.$$

Clearly, $h'(t)=0$ has only one solution: $\bar{t} = \frac{n_1-n_0}{n_0}$. Moreover, we see that $h'(t) > 0$ for $t < \bar{t}$ and $h'(t) < 0$ for $t > \bar{t}$, hence h has its maximum at $\bar{t}$. We observe also that $0 \le \bar{t} \le 1$ by condition (2.2). Evaluating $h(\bar{t})$, we conclude that

$$\max_{0\le t\le 1} h(t) = \frac{1}{s}\sqrt{n_0^2+(n_1-n_0)^2}. \qquad (2.3)$$

Furthermore, since h' has no other zeros, the minimum of h must be attained at one of the boundaries of its domain, i.e.

$$\min_{0\leq t\leq 1} h(t) = \min\{h(0),\ h(1)\} = \frac{1}{s}\min\left\{n_0,\ \frac{1}{2}n_1\sqrt{2}\right\}. \tag{2.4}$$

Interestingly this result for the lim inf is actually due to the following property, which we put in a lemma for future reference. We will later prove a more general version of this lemma.

Lemma 2.1 *If N is a neighbourhood on M_1 of the form* (2.1)*, satisfying $n_0 \leq n_1 \leq 2n_0$, then its induced length function w has the following property:*

$$\liminf_{\|\vec{v}\|\to\infty} \frac{w(\vec{v})}{\|\vec{v}\|} = \frac{1}{s}\min_{\vec{v}\in M_1^*}\frac{N(\vec{v})}{\|\vec{v}\|} = \frac{1}{s}\min\left\{n_0,\ \frac{1}{2}n_1\sqrt{2}\right\}.$$

The following result has thus been established:

Theorem 2.2 *The maximum relative error of an integer neighbourhood N on M_1 with scaling factor s, satisfying $n_0 \leq n_1 \leq 2n_0$, is given by*

$$e = \max\left\{1 - \frac{1}{s}\min\left(n_0,\ \frac{1}{2}n_1\sqrt{2}\right),\ \frac{1}{s}\sqrt{n_0^2 + (n_1 - n_0)^2} - 1\right\}.$$

For instance, the neighbourhood displayed in Figure 1.2(c) has $n_0 = 3$, $n_1 = 4$, $s = 3$, and therefore has a maximum relative error of

$$\max\left\{1 - \frac{2}{3}\sqrt{2},\ \frac{1}{3}\sqrt{10} - 1\right\} = 1 - \frac{2}{3}\sqrt{2} \approx 0.05719, \tag{2.5}$$

which can be compared with the optimal error $e_1^B \approx 0.05505$. We can use the same integer neighbourhood in the C-case, but then we have room to choose an optimal scaling factor. We will return to this in Section 2.3.

2.2 The case $p \geq 2$

We first examine *reduced neighbourhoods* that only permit steps $(\pm p, j)$. Let $p \geq 2$ and define a neighbourhood $\mathcal{N}$ on M_p by

$$\mathcal{N}(i,j) = \begin{cases} n_j & \text{if } |i| = p \text{ and } 0 \leq |j| \leq p \\ \infty & \text{elsewhere} \end{cases} \tag{2.6}$$

where $n_0 \leq n_1 \leq \ldots \leq n_p$ are positive integers. Figure 2.2 shows the general form of a reduced neighbourhood. A scaling factor s is associated to $\mathcal{N}$.

$$
\mathcal{N}: \quad \begin{array}{ccccccc}
n_p & \infty & \cdots & \infty & \cdots & \infty & n_p \\
\vdots & \vdots & \cdots & \vdots & \cdots & \vdots & \vdots \\
n_1 & \infty & \cdots & \infty & \cdots & \infty & n_1 \\
n_0 & \infty & \cdots & \infty & \cdots & \infty & n_0 \\
n_1 & \infty & \cdots & \infty & \cdots & \infty & n_1 \\
\vdots & \vdots & \cdots & \vdots & \cdots & \vdots & \vdots \\
n_p & \infty & \cdots & \infty & \cdots & \infty & n_p
\end{array}
$$

Figure 2.2: General form of a reduced neighbourhood on M_p.

A reduced neighbourhood does not satisfy all the symmetry conditions we imposed up to now: $\mathcal{N}(i,j) = \mathcal{N}(j,i)$ does not hold. Moreover, the induced distance function is not a proper metric, since this "distance" is undefined for all pairs of points $(\vec{u}, \vec{v})$ of $\mathbb{Z}^2$ such that $|u_x - v_x|$ is not a multiple of p. This type of neighbourhood does not have a practical use, but the underlying theory is less complex than for full neighbourhoods, and will be used to generalise the result of Section 2.1.

Just as for $p = 1$, we begin by imposing a condition on the values of $\mathcal{N}$:

$$
n_{j+1} + n_{j-1} \geq 2n_j \qquad (j = 1, \ldots, p-1)\,. \tag{2.7}
$$

This condition is intuitively reasonable, because if it were not satisfied for some j, the step (p, j) would hardly be used. Observe also that (2.7) is satisfied by the true Euclidean length of the corresponding vectors of M_p.

According to the following proposition, condition (2.7) implies that there exists a shortest path from the origin to any point situated in the cone spanned by two subsequent vectors (p, r), $(p, r+1)$ of M_p, that consists of repetitions of these two vectors only.

Proposition 2.3 *Let $p \geq 2$, and let $\mathcal{N}$ be a neighbourhood defined on M_p by* (2.6)*. If the outer values of $\mathcal{N}$ satisfy $n_{j+1}+n_{j-1} \geq 2n_j$ (for $j = 1, \ldots, p-1$) then a shortest path from $(0,0)$ to $(mp, mr+k)$ (where $m, r, k \in \mathbb{Z}_{\geq 0}, 0 \leq r < p$, $0 \leq k < m$) consists of k steps $(p, r+1)$ and $m-k$ steps (p, r).*

Proof. First observe that (2.7) can be rewritten in the following ways:

$$
n_{j+1} - n_j \geq n_j - n_{j-1}, \tag{2.8}
$$

$$
n_{j-1} - n_j \geq n_j - n_{j+1}. \tag{2.9}
$$

Suppose there is a shortest path that contains a step $(p, r+t)$, where $t > 1$. Since the path leads to $(mp, mr+k)$, it also contains a step $(p, r+u)$, with $u \leq 0$. Combined, these steps add a length of $n_{r+t} + n_{r+u}$ to the path. The same distance is covered by a combination of the steps $(p, r+t-1)$ and $(p, r+u+1)$, with total length $n_{r+t-1} + n_{r+u+1}$. From (2.8) we see that

$$n_{r+t} - n_{r+t-1} \geq n_{r+t-1} - n_{r+t-2} \geq \ldots \geq n_{r+1} - n_r$$

and from (2.9) that

$$n_{r+u} - n_{r+u+1} \geq n_{r+u+1} - n_{r+u+2} \geq \ldots \geq n_r - n_{r+1}.$$

But this implies that $n_{r+t} + n_{r+u} \geq n_{r+t-1} + n_{r+u+1}$. So there is a shortest path that does not contain any step $(p, r+t)$ with $t > 1$.

Now suppose this particular shortest path contains a step $(p, r+t)$ with $t < 0$. The path then also contains a step $(p, r+1)$. By a similar argument as before, we find

$$n_{r+t} - n_{r+t+1} \geq \ldots \geq n_r - n_{r+1}.$$

This shows that $n_{r+t} + n_{r+1} \geq n_{r+t+1} + n_r$. Hence, we may replace the steps $(p, r+t)$ and $(p, r+1)$ by a combination of $(p, r+t+1)$ and (p, r). We conclude that there is a shortest path which consists only of steps (p, r) and $(p, r+1)$. □

In order to derive an expression for the maximum relative error of a reduced neighbourhood $\mathcal{N}$ that satisfies (2.7), we impose the following constraint on the values of $\mathcal{N}$:

$$(r+1)n_r > r n_{r+1} \qquad (r = 0, \ldots, p-1)\,. \tag{2.10}$$

Note that this inequality is satisfied by the Euclidean lengths too.

We denote the induced length function of $\mathcal{N}$ by $\mathcal{W}$. The maximum relative error $\mathcal{E}$ of the neighbourhood $\mathcal{N}$ is given by

$$\mathcal{E} = \max\left\{1 - \liminf_{\|\vec{v}\|\to\infty} \frac{\mathcal{W}(\vec{v})}{\|\vec{v}\|},\ \limsup_{\|\vec{v}\|\to\infty} \frac{\mathcal{W}(\vec{v})}{\|\vec{v}\|} - 1\right\} =: \max\left\{\mathcal{E}_{\min},\ \mathcal{E}_{\max}\right\}$$

As usual, we restrict our attention to the first octant of $\mathbb{Z}^2$. For $m, l \in \mathbb{Z}_{\geq 0}$, $l = mr + k$ (where $0 \leq r < p$, $0 \leq k < m$), Proposition 2.3 states that

$$\mathcal{W}(mp, l) = \frac{1}{s}\left\{k n_{r+1} + (m-k) n_r\right\}.$$

Substituting $k = l - mr$, we find

$$\begin{aligned} \mathcal{W}(mp, l) &= \frac{1}{s}\left[(l - mr) n_{r+1} + \left\{m(r+1) - l\right\} n_r\right] \\ &= \frac{1}{s}\left[m\left\{(r+1) n_r - r n_{r+1}\right\} + l\left(n_{r+1} - n_r\right)\right]. \end{aligned}$$

Comparing this with the true Euclidean length of the vector (mp, l), we see that

$$\frac{\mathcal{W}(mp,l)}{\sqrt{(mp)^2+l^2}} = \frac{1}{s}\frac{m\left\{(r+1)n_r - rn_{r+1}\right\} + l\,(n_{r+1}-n_r)}{mp\sqrt{1+(l/mp)^2}}$$

for $mr \le l \le m(r+1)$. By introducing a new variable $t = \frac{l}{mp}$, the previous expression becomes a univariate function, which we call h_r:

$$h_r(t) := \frac{1}{s}\frac{\frac{1}{p}\left\{(r+1)n_r - rn_{r+1}\right\} + (n_{r+1}-n_r)\,t}{\sqrt{1+t^2}} \qquad \left(\frac{r}{p} \le t \le \frac{r+1}{p}\right)$$

for $r = 0, 1, \ldots, p-1$. It is clear that

$$\limsup_{m,l\ge 0} \frac{\mathcal{W}(mp,l)}{\sqrt{(mp)^2+l^2}} = \max_{0\le r<p} \max_{\frac{r}{p}\le t\le \frac{r+1}{p}} h_r(t).$$

By basic calculus (see Appendix A.2) we find that condition (2.10) implies that

$$\max_{0\le r<p} \max_{\frac{r}{p}\le t\le \frac{r+1}{p}} h_r(t) = \frac{1}{s}\max_{0\le r\le p-1} H_r, \tag{2.11}$$

where H_r is given by

$$H_r = \begin{cases} \sqrt{\frac{1}{p^2}\left\{(r+1)n_r - rn_{r+1}\right\}^2 + (n_{r+1}-n_r)^2} & \text{if } \left\lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \right\rfloor = r \\ \frac{n_r}{\sqrt{p^2+r^2}} & \text{if } \left\lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \right\rfloor < r \\ \frac{n_{r+1}}{\sqrt{p^2+(r+1)^2}} & \text{if } \left\lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \right\rfloor > r \end{cases}$$

We have thus established a simple (if slightly tedious) method for determining $\mathcal{E}_{\max}$. It turns out that $\mathcal{E}_{\min}$ can be found with less work because of the following lemma. This is an analogue to Lemma 2.1 in the general case.

Lemma 2.4 *Let $p \ge 2$ and let $\mathcal{N}$ be a neighbourhood of the form* (2.6)*, satisfying $n_{j+1} + n_{j-1} \ge 2n_j$ (for $j = 1, \ldots, p-1$), defined on M_p. The induced distance function $\mathcal{W}$ has the following property:*

$$\liminf_{\|\vec{v}\|\to\infty} \frac{\mathcal{W}(\vec{v})}{\|\vec{v}\|} = \frac{1}{s}\min_{\vec{v}\in M_p^*} \frac{\mathcal{N}(\vec{v})}{\|\vec{v}\|} = \frac{1}{s}\min_{0\le k\le p} \frac{\mathcal{N}(p,k)}{\sqrt{p^2+k^2}}.$$

Proof. The second equality is trivial. We give a proof of the first equality. Let $\mu = \min\limits_{\vec{v}\in M_p^*} \frac{\mathcal{N}(\vec{v})}{\|\vec{v}\|} = \frac{\mathcal{N}(p,u)}{\sqrt{p^2+u^2}}$ for a certain $0 \le u \le p$. It is clear that

$$\liminf_{\|\vec{v}\|\to\infty} \frac{\mathcal{W}(\vec{v})}{\|\vec{v}\|} \le \liminf_{m\ge 0} \frac{\mathcal{W}(mp,mu)}{\sqrt{(mp)^2+(mu)^2}} = \frac{1}{s}\liminf_{m\ge 0} \frac{m\mathcal{N}(p,u)}{m\sqrt{p^2+u^2}} = \frac{\mu}{s}.$$

On the other hand, suppose that a shortest path from $(0,0)$ to (mp,l) consists of steps (p,j_r) (with $r=1,\ldots,m$). Then

$$\frac{\mathcal{W}(mp,l)}{\sqrt{(mp)^2+l^2}} = \frac{1}{s}\frac{\sum_{r=1}^m \mathcal{N}(p,j_r)}{\sqrt{(mp)^2+l^2}}.$$

The sum can be bounded from below by

$$\sum_{r=1}^m \mathcal{N}(p,j_r) = \sum_{r=1}^m \frac{\mathcal{N}(p,j_r)}{\sqrt{p^2+j_r^2}}\sqrt{p^2+j_r^2} \geq \mu \sum_{r=1}^m \sqrt{p^2+j_r^2}.$$

Thus

$$\liminf_{\|\vec{v}\|\to\infty} \frac{\mathcal{W}(\vec{v})}{\|\vec{v}\|} \geq \frac{1}{s} \liminf_{m\geq 0, l\geq 0} \frac{\mu\sum_{r=1}^m \sqrt{p^2+j_r^2}}{\sqrt{(mp)^2+l^2}} \geq \frac{\mu}{s},$$

since the total Euclidean length of the shortest path can never be smaller than the Euclidean length of the vector (mp,l). □

The following result has thus been established:

Proposition 2.5 *The maximum relative error of a reduced neighbourhood* $\mathcal{N}$ *of the form* (2.6) *satisfying* $n_{j+1}+n_{j-1}\geq 2n_j$ *(for* $j=1,\ldots,p-1$*) and* $(r+1)n_r > rn_{r+1}$ *(for* $r=0,\ldots,p-1$*) is*

$$\mathcal{E} = \max\left\{1-\frac{1}{s}\min_{0\leq k\leq p}\frac{\mathcal{N}(p,k)}{\sqrt{p^2+k^2}},\ \frac{1}{s}\max_{0\leq r\leq p-1} H_r - 1\right\},$$

with H_r *given by*

$$H_r = \begin{cases} \sqrt{\frac{1}{p^2}\left\{(r+1)n_r - rn_{r+1}\right\}^2 + (n_{r+1}-n_r)^2} & \text{if } \left\lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \right\rfloor = r \\ \frac{n_r}{\sqrt{p^2+r^2}} & \text{if } \left\lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \right\rfloor < r \\ \frac{n_{r+1}}{\sqrt{p^2+(r+1)^2}} & \text{if } \left\lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \right\rfloor > r \end{cases}$$

Reduced neighbourhoods have no practical use, and Proposition 2.5 can only be used as a stepping stone for finding the maximum relative error of a general integer neighbourhood. To achieve this we need the following proposition.

Proposition 2.6 *Let* $p\geq 2$*, let* $n_0\leq n_1\leq\ldots\leq n_p$ *satisfy* $n_{j+1}+n_{j-1}\geq 2n_j$ *(for* $j=1,\ldots,p-1$*) and let* $\mathcal{N}$ *be defined by* (2.6)*, with scaling factor* s*. Let* N *be an integer neighbourhood on* M_p *with the same scaling factor* s*, such that* $N(\pm p,j)=\mathcal{N}(\pm p,j)$ *for all* $|j|\leq p$*, and such that moreover*

$$\frac{N(i,j)}{\sqrt{i^2+j^2}} \geq \min_{0\leq k\leq p}\frac{N(p,k)}{\sqrt{p^2+k^2}} \qquad \left((i,j)\in M_p^*\right). \tag{2.12}$$

If e *and* $\mathcal{E}$ *are the maximum relative errors of* N *and* $\mathcal{N}$*, respectively, then* $e\leq\mathcal{E}$*.*

Proof. Let w and $\mathcal{W}$ be the length functions induced by N and $\mathcal{N}$, respectively. It suffices to prove that both

$$\liminf_{\|\vec{v}\|\to\infty} \frac{w(\vec{v})}{\|\vec{v}\|} \geq \liminf_{\|\vec{v}\|\to\infty} \frac{\mathcal{W}(\vec{v})}{\|\vec{v}\|} \quad \text{and} \quad \limsup_{\|\vec{v}\|\to\infty} \frac{w(\vec{v})}{\|\vec{v}\|} \leq \limsup_{\|\vec{v}\|\to\infty} \frac{\mathcal{W}(\vec{v})}{\|\vec{v}\|}$$

hold. We start with the $\liminf$.

Suppose a shortest path from $(0,0)$ to (m,l) consists of steps (i_r, j_r). From (2.12) we know that

$$\sum_r N(i_r, j_r) \geq \left(\min_{0\leq k\leq p} \frac{N(p,k)}{\sqrt{p^2+k^2}} \right) \sum_r \sqrt{i_r^2 + j_r^2}.$$

Using Lemma 2.4, this means that

$$\begin{aligned}
\liminf_{\|\vec{v}\|\to\infty} \frac{w(\vec{v})}{\|\vec{v}\|} &\geq \frac{1}{s} \liminf_{m,l\geq 0} \left(\min_{0\leq k\leq p} \frac{N(p,k)}{\sqrt{p^2+k^2}} \right) \frac{\sum_r \sqrt{i_r^2+j_r^2}}{\sqrt{m^2+l^2}} \\
&\geq \frac{1}{s} \min_{0\leq k\leq p} \frac{N(p,k)}{\sqrt{p^2+k^2}} = \liminf_{\|\vec{v}\|\to\infty} \frac{\mathcal{W}(\vec{v})}{\|\vec{v}\|}.
\end{aligned}$$

Now let $(mp+u, l) \in \mathbb{Z}^2$, with $m > 0$ and $0 < u \leq p$. One way – not necessarily the shortest – to get from $(0,0)$ to $(mp+u,l)$ is the following: first take a shortest path from $(0,0)$ to (mp,l) and then make one step $(u,0)$. This implies that $w(mp+u,l) \leq w(mp,l) + \frac{1}{s}N(u,0)$. Obviously, $w(mp,l) \leq \mathcal{W}(mp,l)$. We find that

$$\begin{aligned}
\limsup_{m,l\geq 0} \frac{w(mp+u,l)}{\|(mp+u,l)\|} &\leq \limsup_{m,l\geq 0} \frac{\mathcal{W}(mp,l) + \frac{1}{s}N(u,0)}{\|(mp+u,l)\|} \\
&\leq \limsup_{m,l\geq 0} \frac{\mathcal{W}(mp,l) + \frac{1}{s}N(u,0)}{\|(mp,l)\|} = \limsup_{m,l\geq 0} \frac{\mathcal{W}(mp,l)}{\|(mp,l)\|}
\end{aligned}$$

since $\frac{1}{s}N(u,0)$ is bounded and does not affect the limit. It follows that $\limsup_{\|\vec{v}\|\to\infty} \frac{w(\vec{v})}{\|\vec{v}\|} \leq \limsup_{\|\vec{v}\|\to\infty} \frac{\mathcal{W}(\vec{v})}{\|\vec{v}\|}$. We conclude that $e \leq \mathcal{E}$. □

Proposition 2.6 states that the maximum relative error of an integer neighbourhood satisfying (2.12) is bounded by the maximum relative error of the associated reduced neighbourhood that only contains its outer values. The latter maximum relative error can be evaluated easily using Proposition 2.5 provided the reduced neighbourhood satisfies (2.7) and (2.10). We thus have obtained the following result.

Theorem 2.7 *Let $p \geq 2$ and let N be an integer neighbourhood on M_p with scaling factor s. Write $n_j = N(p,j)$ for $j = 0, \ldots, p$. If the inequalities* *(i)* $n_0 \le n_1 \le \ldots \le n_p$,

(ii) $n_{j+1} + n_{j-1} \ge 2n_j$ *(for* $j = 1, \ldots, p-1$*)*,

(iii) $(r+1)n_r > rn_{r+1}$ *(for* $r = 0, \ldots, p-1$*)*,

are all satisfied and if moreover

$$\frac{N(i,j)}{\sqrt{i^2+j^2}} \ge \min_{0\le k\le p} \frac{n_k}{\sqrt{p^2+k^2}}$$

holds for all $(i,j) \in M_p^*$*, then the maximum relative error satisfies*

$$e \le \max\left\{1 - \frac{1}{s}\min_{0\le k\le p}\frac{n_k}{\sqrt{p^2+k^2}},\ \frac{1}{s}\max_{0\le r\le p-1} H_r - 1\right\}, \tag{2.13}$$

where H_r *is given by*

$$H_r = \begin{cases} \sqrt{\frac{1}{p^2}\left\{(r+1)n_r - rn_{r+1}\right\}^2 + (n_{r+1}-n_r)^2} & \text{if } \lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \rfloor = r \\ \frac{n_r}{\sqrt{p^2+r^2}} & \text{if } \lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \rfloor < r \\ \frac{n_{r+1}}{\sqrt{p^2+(r+1)^2}} & \text{if } \lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \rfloor > r \end{cases}$$

A slightly shorter formulation of this result can be obtained by imposing one additional restriction on the outer elements of N. We have:

Corollary 2.8 *Let* $p \ge 2$ *and let* N *be an integer neighbourhood on* M_p *with scaling factor* s *that satisfies all the inequalities of Theorem* 2.7*. If it also holds that*

$$1 + \frac{r}{p^2+r^2} \le \frac{n_{r+1}}{n_r} < 1 + \frac{r+1}{p^2+r(r+1)} \tag{2.14}$$

for $r = 0, 1, \ldots, p-1$*, then the maximum relative error satisfies* (2.13) *with* $H_r = \sqrt{\frac{1}{p^2}\left\{(r+1)n_r - rn_{r+1}\right\}^2 + (n_{r+1}-n_r)^2}$.

Proof. The right-hand inequality in (2.14) is $\frac{n_{r+1}}{n_r} < \frac{p^2+(r+1)^2}{p^2+r(r+1)}$. This can be rewritten as $n_{r+1}\left\{p^2 + r(r+1)\right\} < n_r\left\{p^2+(r+1)^2\right\}$, which in turn is equivalent to $p^2(n_{r+1}-n_r) < (r+1)\left\{(r+1)n_r - rn_{r+1}\right\}$, i.e.

$$\frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} < r+1.$$

Similarly, the left-hand inequality in (2.14) is equivalent to

$$\frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \ge r.$$

The result now follows from the definition of H_r in Theorem 2.7. □

Theorem 2.7 only provides an upper bound on the maximum relative error. While it seems unlikely that the maximum relative error of every integer neighbourhood is determined solely by its outer values, an examination of the literature shows that equality holds for most previously published neighbourhoods. It should be noted however that virtually all neighbourhoods considered in the literature are of size $p \leq 3$. For larger p the number of inner points in M_p increases rapidly, and it becomes more and more likely that the maximum relative error of an integer neighbourhood is strictly smaller than the error of its associated reduced neighbourhood.

Anyone who wants to attempt to proof that equality holds in (2.13) should also be aware that even for the true Euclidean distance it is *not* true that we can always find a shortest path to any point (mp, k) that only contains steps of the form (p, j). The following counter-example exists for $p = 3$: if we use only the outer steps, the shortest path from $(0, 0)$ to $(6, 3)$ is $[(3, 1); (3, 2)]$ with total Euclidean length $\sqrt{10} + \sqrt{13} \approx 6.77$, whereas the overall shortest path is $[(2, 1); (2, 1); (2, 1)]$ with length $3\sqrt{5} \approx 6.71$.

On the other hand, it is interesting to note that equality *does* hold in (2.13) for the optimal neighbourhoods w_p^B, w_p^C and w_p^D given in Section 1.4.[1] A proof can be found in Appendix A.3. This suggests that we may at least hope for equality for all integer neighbourhoods for which the approximation to the optimal neighbourhood is "good enough".

2.3 The rôle of the scaling factor

The maximum relative error of an integer neighbourhood can not be determined without prescribing a scaling factor s. We still have to decide which scaling factor to use. In the B- and D-cases there is no choice: the restrictions require that $s = N(1, 0)$. In the unrestricted C-case, however, it is not required that any vector of M_p be given the true Euclidean distance, and we can optimise the scaling factor. The optimal scaling factor was proposed by Verwer in [22], although the concept was introduced by Vossepoel in [23].

Let p be any positive integer. Suppose we are given an integer neighbourhood N on M_p with a maximum relative error (less than or) equal to $\max\{e_{\min},\ e_{\max}\}$. Looking back to Theorem 2.2 and Theorem 2.7, in both cases the expression for the maximum relative error is of the following form:

$$e_{\min} = 1 - \frac{c_{\min}}{s},\ e_{\max} = \frac{c_{\max}}{s} - 1.$$

If we now put

$$s := \frac{1}{2}\left(c_{\max} + c_{\min}\right) \tag{2.15}$$

[1]Of course, as it stands we can not use (2.13) for non-integer neighbourhoods such as w_p^B. The reader is referred to Appendix A.3 for details.

for the scaling factor, this will yield a maximum relative error of

$$1 - \frac{c_{\min}}{s} = \frac{c_{\max}}{s} - 1 = \frac{c_{\max} - c_{\min}}{c_{\max} + c_{\min}}. \tag{2.16}$$

The maximal deviations from the true Euclidean distance in the positive ($c_{\max}$) and negative direction ($c_{\min}$) are now equal, and this is the lowest possible maximum relative error for this integer neighbourhood. (The importance of symmetry in $c_{\min}$ and $c_{\max}$ is clear from looking at the D-case, where the error is intentionally made very unsymmetric by demanding that $c_{\min} = s$. As was seen in Table 1.1, the resulting maximum relative errors are much higher than in the unrestricted case.)

As an example for $p = 1$, consider the neighbourhood displayed in Figure 1.2(c). According to (2.5) we have $c_{\min} = 2\sqrt{2}$ and $c_{\max} = \sqrt{10}$. Expression (2.15) gives

$$s = \frac{1}{2}\left(\sqrt{10} + 2\sqrt{2}\right) \approx 2.99535.$$

Moreover, it follows from (2.16) that the resulting neighbourhood $w = \frac{N}{s}$ has a maximum relative error of $\frac{\sqrt{10}-2\sqrt{2}}{\sqrt{10}+2\sqrt{2}} \approx 0.05573$. This is an improvement of the original choice of scaling factor $s = 3$, which yielded an error of 0.05719.

Chapter 3

Construction of integer neighbourhoods: part one

In this chapter and the next one, classes of integer neighbourhoods will be constructed that approximate the optimal neighbourhoods given by the expressions (1.13)–(1.15). We will take care to ensure that all the conditions introduced in the previous chapter are satisfied by these integer neighbourhoods, so Theorem 2.2 and Theorem 2.7 can be used to calculate (for $p = 1$) or bound (for $p \geq 2$) the maximum relative errors.

3.1 The case $p = 1$

We first construct integer neighbourhoods on M_1 that satisfy constraint (2.2) and that are approximations of the optimal neighbourhoods w_1^B, w_1^C and w_1^D. For this we need the following auxiliary lemma.

Lemma 3.1 *Let $\frac{1}{2}\sqrt{2} < c \leq 1$ and let n be a positive integer. The following choices of n_0 and n_1 all satisfy $n_0 \leq n_1 \leq 2n_0$:*

(i) $n_0 = n$, $n_1 = \langle nc\sqrt{2} \rangle$;

(ii) $n_0 = \langle nc \rangle$, $n_1 = \langle nc\sqrt{2} \rangle$;

(iii) $n_0 = n$, $n_1 = \lceil n\sqrt{2} \rceil$.

Proof. (i): It follows from $c > \frac{1}{2}\sqrt{2}$ that $\langle nc\sqrt{2} \rangle \geq \langle n \rangle = n$, so $n_0 \leq n_1$ is satisfied. And it follows from $c \leq 1$ that $\langle nc\sqrt{2} \rangle \leq \langle n\sqrt{2} \rangle \leq \langle 2n \rangle = 2n$, so $n_1 \leq 2n_0$ also holds.

(ii): It is obvious that $n_1 \geq n_0$ for every n. First assume that $n \geq 4$. Using that $x - \frac{1}{2} < \langle x \rangle \leq x + \frac{1}{2}$, we find that

$$2\langle nc \rangle - \langle nc\sqrt{2} \rangle > nc\left(2 - \sqrt{2}\right) - \frac{3}{2} > n\left(\sqrt{2} - 1\right) - \frac{3}{2} \geq 4\sqrt{2} - \frac{11}{2} > 0,$$

which shows that $2n_0 \geq n_1$. For $n = 3$ we have:

$$n_0 = \langle 3c \rangle = \begin{cases} 2 & \text{if } \frac{1}{2}\sqrt{2} < c < \frac{5}{6} \\ 3 & \text{if } \frac{5}{6} \leq c \leq 1 \end{cases}$$

$$n_1 = \langle 3c\sqrt{2} \rangle = \begin{cases} 3 & \text{if } \frac{1}{2}\sqrt{2} < c < \frac{7}{6\sqrt{2}} \\ 4 & \text{if } \frac{7}{6\sqrt{2}} \leq c \leq 1 \end{cases}$$

and it is easy to see that $n_1 \leq 2n_0$ is always satisfied for these values. The cases $n = 1$ and $n = 2$ can be checked similarly.

(iii): $n_1 \geq n_0$ trivially holds for every n. Moreover, $\lceil n\sqrt{2} \rceil \leq \lceil 2n \rceil = 2n$ shows that $n_1 \leq 2n_0$ also holds for every n. □

Definition 3.2 *Let n be a positive integer. The following integer neighbourhoods are defined on M_1:*

$$_nN_1^B(i,j) := \begin{cases} n & \text{if } |i| + |j| = 1 \\ \langle n\left(1 - e_1^B\right)\sqrt{2} \rangle & \text{if } |i| + |j| = 2 \end{cases}$$

$$_nN_1^C(i,j) := \begin{cases} \langle n\left(1 - e_1^C\right) \rangle & \text{if } |i| + |j| = 1 \\ \langle n\left(1 - e_1^C\right)\sqrt{2} \rangle & \text{if } |i| + |j| = 2 \end{cases}$$

$$_nN_1^D(i,j) := \begin{cases} n & \text{if } |i| + |j| = 1 \\ \lceil n\sqrt{2} \rceil & \text{if } |i| + |j| = 2 \end{cases}$$

with $e_1^B = 3 - \sqrt{2} - 2\sqrt{2 - \sqrt{2}}$ *and* $e_1^C = \frac{\sqrt{4-2\sqrt{2}}-1}{\sqrt{4-2\sqrt{2}}+1}$.

Proposition 3.3 *For $X \in \{B, C, D\}$, the maximum relative error of $_nN_1^X$ is*

$$_ne_1^X = \max\left\{1 - \frac{1}{s}\min\left(n_0,\ \frac{1}{2}n_1\sqrt{2}\right),\ \frac{1}{s}\sqrt{n_0^2 + (n_1 - n_0)^2} - 1\right\},$$

with $n_0 = {}_nN_1^X(1,0)$, $n_1 = {}_nN_1^X(1,1)$.

Proof. The result follows immediately from Lemma 3.1 and Theorem 2.2 by choosing c appropriately. □

As an example, $_3N_1^B$ takes the values $n_0 = 3$, $n_1 = \langle 4.0091 \rangle = 4$ and the scaling factor $s = n_0 = 3$. This is in fact the neighbourhood displayed in Figure 1.2(c), which was originally published by Borgefors.

The neighbourhoods constructed in Definition 3.2 are integer approximations to the optimal neighbourhoods given in Section 1.4. Moreover, we have a simple expression for the maximum relative error of these neighbourhoods. By varying the choice of n, good integer approximations may now be found in the case $p = 1$. The results of these experiments are deferred until Chapter 5.

3.2 The case $p \geq 2$

Following the structure of the previous chapter, we begin with constructing reduced neighbourhoods that satisfy (2.7) and (2.10). The other neighbourhood values will then be filled in in such a way that condition (2.12) is not violated. The resulting integer neighbourhoods then satisfy all conditions of Theorem 2.7.

We begin by proving an auxiliary lemma.

Lemma 3.4 *Let $a \neq 0$. The function*

$$g_a(x) := \sqrt{a^2 + (x+1)^2} - 2\sqrt{a^2 + x^2} + \sqrt{a^2 + (x-1)^2}$$

is monotone decreasing to 0 *on* $\left[\frac{1}{2}, \infty\right)$.

Proof. We have

$$g_a'(x) = \left(\frac{x+1}{\sqrt{a^2 + (x+1)^2}} - \frac{x}{\sqrt{a^2 + x^2}}\right) - \left(\frac{x}{\sqrt{a^2 + x^2}} - \frac{x-1}{\sqrt{a^2 + (x-1)^2}}\right).$$

Put

$$k_a(y) := \frac{y+1}{\sqrt{a^2 + (y+1)^2}} - \frac{y}{\sqrt{a^2 + y^2}}.$$

Then

$$k_a'(y) = \frac{a^2}{(a^2 + (y+1)^2)^{3/2}} - \frac{a^2}{(a^2 + y^2)^{3/2}} < 0 \qquad \left(y > -\frac{1}{2}\right).$$

Hence $k_a(x) < k_a(x-1)$ for all $x \geq \frac{1}{2}$, and thus

$$g_a'(x) = k_a(x) - k_a(x-1) < 0 \qquad \left(x \geq \frac{1}{2}\right).$$

It is clear that $\lim_{x \to \infty} g_a(x) = 0$. □

Lemma 3.5 *Let $p \geq 2$ and $\frac{p}{\sqrt{p^2+1}} < c \leq 1$, and let n be a positive integer. The following choices of $n_0, \ldots, n_p$ all satisfy $n_0 \leq n_1 \leq \ldots \leq n_p$ and $n_{j+1} + n_{j-1} \geq 2n_j$ (for $j = 1, \ldots, p-1$):*

(i) $n_0 = np$, $n_j = \langle nc\sqrt{p^2 + j^2} \rangle$ $(j = 1, \ldots, p)$, *for* $n > \frac{2}{cg_p(p-1)}$;

(ii) $n_j = \langle nc\sqrt{p^2 + j^2} \rangle$ $(j = 0, \ldots, p)$, *for* $n > \frac{2}{cg_p(p-1)}$;

(iii) $n_j = \lceil n\sqrt{p^2 + j^2} \rceil$ $(j = 0, \ldots, p)$, *for* $n > \frac{2}{g_p(p-1)}$.

Proof. (i): It is clear that $n_1 \leq n_2 \leq \ldots \leq n_p$. From $c > \frac{p}{\sqrt{p^2+1}}$ it follows that $n_1 = \langle nc\sqrt{p^2+1}\rangle \geq \langle np\rangle = n_0$. It remains to be shown that (2.7) holds for all $j = 1, \ldots, p-1$.

First take $j > 1$. From $x - \frac{1}{2} < \langle x\rangle \leq x + \frac{1}{2}$ we see that

$$\langle nc\sqrt{p^2+(j+1)^2}\rangle - 2\langle nc\sqrt{p^2+j^2}\rangle + \langle nc\sqrt{p^2+(j-1)^2}\rangle > ncg_p(j) - 2.$$

By Lemma 3.4 the term $g_p(j)$ is positive for all j. Moreover, the smallest value is achieved for the largest j, i.e. for $j = p-1$. So

$$ncg_p(j) \geq ncg_p(p-1) > 2$$

by our assumption on n. This shows that (2.7) holds for $j > 1$.

For $j = 1$ we have

$$\langle nc\sqrt{p^2+4}\rangle - 2\langle nc\sqrt{p^2+1}\rangle + np > nc\left(\sqrt{p^2+4} - 2\sqrt{p^2+1}\right) + np - \frac{3}{2}.$$

It follows from $c \leq 1$ that

$$nc\left(\sqrt{p^2+4} - 2\sqrt{p^2+1}\right) + np \geq ncg_p(1).$$

By the assumption on n, $ncg_p(1)$ is larger than $\frac{2g_p(1)}{g_p(p-1)}$. But we know from Lemma 3.4 that $g_p(1) \geq g_p(p-1)$, which shows that $ncg_p(1) - \frac{3}{2}$ is positive, as required.

(ii): $n_0 \leq n_1 \leq \ldots \leq n_p$ trivially holds. The remainder of the proof is completely analogous to (i).

(iii): Again, $n_0 \leq n_1 \leq \ldots \leq n_p$ holds trivially. The proof of (2.7) is similar to (i), but this time we use that $x \leq \lceil x\rceil < x + 1$. □

Just as we did for $p = 1$ in Definition 3.2, we would now like to use these three results to get approximating integer neighbourhoods to w_p^B, w_p^C and w_p^D, respectively – albeit only reduced neighbourhoods for now –, by substituting $c = 1 - e_p^B$ in Lemma 3.5(i), $c = 1 - e_p^C$ in (ii), and by directly applying (iii). This construction puts a lower bound on n in all three cases, namely $n > n_p^B$ in the B-case, where

$$n_p^B = \frac{2p^2}{\left(p\sqrt{p^2+1} + 2\sqrt{p^2+1-p\sqrt{p^2+1}} - 2\right) g_p(p-1)},$$

and similar expressions for $n > n_p^C$ in the C-case and $n > n_p^D$ in the D-case. The lower bound for n in the B-case is shown in the second column of Table 3.1 for $2 \leq p \leq 10$.

Table 3.1: Theoretical and practical lower bounds on n in Lemma 3.5(i), when applied to the B-case, for $2 \leq p \leq 10$. The second column gives the theoretical bounds, the third column shows *all* values of n for which choosing $n_0 = np$, $n_j = \langle n\left(1 - e_p^B\right)\sqrt{p^2 + j^2}\rangle$ $(j = 1, \ldots, p)$ gives a reduced neighbourhood that satisfies (2.7).

p	n_p^B	OK for
2	5.72	$n \geq 1$
3	10.41	$n \geq 2$
4	15.59	$n \geq 1$
5	20.95	$n = 2, 3$ and for $n \geq 5$
6	26.42	$n = 4, 6$ and for $n \geq 8$
7	31.94	$n = 3, 4, 5$ and for $n \geq 7$
8	37.49	$n \geq 2$
9	43.07	$n = 2, 5, 6, 7, 8$ and for $n \geq 10$
10	48.66	$n = 4, 5, 6, 9, 10, 11$ and for $n \geq 14$

For small values of p, it is easy to check whether a given reduced neighbourhood satisfies the $p-1$ inequalities (2.7). By doing this for all values $1 \leq n < n_p^B$, the lower bound can be sharpened.

The third column of Table 3.1 shows, for $2 \leq p \leq 10$, the values of n for which choosing $n_0 = np$, $n_j = \langle n\left(1 - e_p^B\right)\sqrt{p^2 + j^2}\rangle$ $(j = 1, \ldots, p)$ yields a reduced neighbourhood that satisfies (2.7). This demonstrates – for small values of p, anyway – that (2.7) is a mild condition on neighbourhoods of this form.

We use the same approach in the C- and D-cases; the results are given in Tables 3.2 and 3.3.

The following proposition summarises this result.

Proposition 3.6 *Let $2 \leq p \leq 10$ and n a positive integer. The following reduced neighbourhoods on M_p satisfy $n_{j+1} + n_{j-1} \geq 2n_j$ (for $j = 1, \ldots, p-1$):*

$$
{}_n\mathcal{N}_p^B(i,j) := \begin{cases} np & \text{if } |i| = p \text{ and } j = 0 \\ \langle n\left(1 - e_p^B\right)\sqrt{p^2 + j^2}\rangle & \text{if } |i| = p \text{ and } 1 \leq |j| \leq p \\ \infty & \text{elsewhere} \end{cases}
$$

for the values of n displayed in Table 3.1;

$$
{}_n\mathcal{N}_p^C(i,j) := \begin{cases} \langle n\left(1 - e_p^C\right)\sqrt{p^2 + j^2}\rangle & \text{if } |i| = p \text{ and } 0 \leq |j| \leq p \\ \infty & \text{elsewhere} \end{cases}
$$

for the values of n displayed in Table 3.2;

$$
{}_n\mathcal{N}_p^D(i,j) := \begin{cases} \lceil n\sqrt{p^2 + j^2}\rceil & \text{if } |i| = p \text{ and } 0 \leq |j| \leq p \\ \infty & \text{elsewhere} \end{cases}
$$

Table 3.2: Practical lower bounds on n in Lemma 3.5(ii), when applied to the C-case, for $2 \leq p \leq 10$. The values of n are indicated for which choosing $n_j = \langle n \left(1 - e_p^C\right) \sqrt{p^2 + j^2} \rangle$ $(j = 0, \ldots, p)$ gives a reduced neighbourhood that satisfies (2.7).

p	OK for
2	$n \geq 1$
3	$n \geq 2$
4	$n \geq 1$
5	$n = 2, 3$ and for $n \geq 5$
6	$n = 4, 6$ and for $n \geq 8$
7	$n = 3, 4, 5$ and for $n \geq 7$
8	$n \geq 2$
9	$n = 2, 5, 6, 7, 8, 10, 11, 12, 13, 15$ and for $n \geq 17$
10	$n = 4, 5, 6, 9, 10, 11$ and for $n \geq 13$

Table 3.3: Practical lower bounds on n in Lemma 3.5(iii), when applied to the D-case, for $2 \leq p \leq 10$. The values of n are indicated for which choosing $n_j = \lceil n\sqrt{p^2 + j^2} \rceil$ $(j = 0, \ldots, p)$ yields a reduced neighbourhood that satisfies (2.7).

p	OK for
2	$n \geq 2$
3	$n \geq 2$
4	$n \geq 3$
5	$n \geq 3$
6	$n \geq 5$
7	$n \geq 4$
8	$n = 6$ and for $n \geq 9$
9	$n = 5$ and for $n \geq 8$
10	$n \geq 8$

for the values of n *displayed in Table* 3.3.

Since we want to use Proposition 2.5 to calculate the maximum relative errors of these neighbourhoods, we must check that condition (2.10) holds. The following lemma implies that. Maple software was used for certain parts of the proof.

Lemma 3.7 *Let* $p \geq 2$. *For every* $n \geq 1$ *the neighbourhoods* ${}_n\mathcal{N}_p^B$, ${}_n\mathcal{N}_p^C$ *and* ${}_n\mathcal{N}_p^D$ *defined in Proposition* 3.6 *satisfy* $(r+1)n_r > rn_{r+1}$ *(for* $r = 0, \ldots, p-1$*).*

Proof. The proof is given in Appendix A.4. □

This concludes our work on reduced neighbourhoods. We will now extend the reduced neighbourhoods ${}_n\mathcal{N}_p^B$, ${}_n\mathcal{N}_p^C$ and ${}_n\mathcal{N}_p^D$ to proper integer neighbourhoods by filling in the inner values in such a way that they approximate the corresponding values of w_p^B (resp. w_p^C, w_p^D) *and* that (2.12) is not violated. (Obviously there is a trade-off here and the approximation is not as good as for the outer values. This is not a problem, because we found in Proposition 2.6 that the maximum relative error is dominated by the outer values anyway.) Approximating integer neighbourhoods are thus found with maximum relative errors that are guaranteed not to exceed a known value.

Definition 3.8 *Let* $2 \leq p \leq 10$ *and* n *a positive integer. The following neighbourhoods are defined on* M_p*:*

$$
{}_nN_p^B(i,j) := \begin{cases} n|i| & \text{if } 1 \leq |i| \leq p \text{ and } j = 0 \\ n|j| & \text{if } 1 \leq |j| \leq p \text{ and } i = 0 \\ \langle n\left(1 - e_p^B\right)\sqrt{p^2 + j^2}\rangle & \text{if } |i| = p \text{ and } 1 \leq |j| \leq p \\ \lceil \mu\sqrt{i^2 + j^2}\rceil & \text{for all other vectors in } M_p^* \end{cases}
$$

where $\mu = \min\limits_{0 \leq k \leq p} \frac{{}_nN_p^B(p,k)}{\sqrt{p^2+k^2}}$, *for the values of* n *displayed in Table* 3.1*;*

$$
{}_nN_p^C(i,j) := \begin{cases} \langle n\left(1 - e_p^C\right)\sqrt{p^2 + j^2}\rangle & \text{if } |i| = p \text{ and } 0 \leq |j| \leq p \\ \lceil \mu\sqrt{i^2 + j^2}\rceil & \text{for all other vectors in } M_p^* \end{cases}
$$

where $\mu = \min\limits_{0 \leq k \leq p} \frac{{}_nN_p^C(p,k)}{\sqrt{p^2+k^2}}$, *for the values of* n *displayed in Table* 3.2*;*

$$
{}_nN_p^D(i,j) := \begin{cases} n|i| & \text{if } 1 \leq |i| \leq p \text{ and } j = 0 \\ n|j| & \text{if } 1 \leq |j| \leq p \text{ and } i = 0 \\ \lceil n\sqrt{i^2 + j^2}\rceil & \text{for all other vectors in } M_p^* \end{cases}
$$

for the values of n *displayed in Table* 3.3.
The values of e_p^B *and* e_p^C *are given by* (1.10) *and* (1.12).

Proposition 3.9 *For $X \in \{B, C\}$, the maximum relative error of ${}_nN_p^X$ satisfies*

$$ {}_ne_p^X \leq \max\left\{1 - \frac{1}{s}\min_{0\leq k\leq p}\frac{{}_nN_p^X(p,k)}{\sqrt{p^2+k^2}},\ \frac{1}{s}\max_{0\leq r\leq p-1} H_r - 1\right\} $$

where, putting $n_j = {}_nN_p^X(p,j)$,

$$ H_r = \begin{cases} \sqrt{\frac{1}{p^2}\left\{(r+1)n_r - rn_{r+1}\right\}^2 + (n_{r+1}-n_r)^2} & \text{if } \lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \rfloor = r \\ \frac{n_r}{\sqrt{p^2+r^2}} & \text{if } \lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \rfloor < r \\ \frac{n_{r+1}}{\sqrt{p^2+(r+1)^2}} & \text{if } \lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \rfloor > r \end{cases} $$

The maximum relative error of ${}_nN_p^D$ satisfies

$$ {}_ne_p^D \leq \frac{1}{s}\max_{0\leq r\leq p-1} H_r - 1 $$

with H_r as given above, putting $n_j = {}_nN_p^D(p,j)$.

Proof. We give the proof for ${}_nN_p^B$. Condition (2.12) is satisfied, because by definition

$$ {}_nN_p^B(i,j) = \lceil \mu\sqrt{i^2+j^2}\rceil \geq \mu\sqrt{i^2+j^2} $$

holds for every $(i,j) \in M_p^*$ with $i \neq 0$ and $j \neq 0$, and

$$ \min_{0\leq k\leq p}\frac{{}_nN_p^B(p,k)}{\sqrt{p^2+k^2}} \leq \frac{{}_nN_p^B(p,0)}{p} = n $$

shows that (2.12) is also true for every $(0,j)$ and $(i,0)$. The result now follows from Proposition 3.6, Lemma 3.7 and Theorem 2.7.

The proofs for ${}_nN_p^C$ and ${}_nN_p^D$ are analogous. The term involving the minimum may be dropped for ${}_ne_p^D$ because

$$ \frac{1}{s}\min_{0\leq k\leq p}\frac{{}_nN_p^D(p,k)}{\sqrt{p^2+k^2}} = \frac{n}{s} = 1, $$

where we use that $s = n$ in the D-case. □

As an example, we construct the neighbourhood ${}_5N_2^B$ by calculating its values on the first octant. To improve readability, we drop all the indices here and just write N. Since we are in the B-case, we have $N(1,0) = 5$ and $N(2,0) = 10$. The other outer values are $N(2,1) = \langle 10.9713\rangle = 11$ and $N(2,2) = \langle 13.8778\rangle = 14$. The only remaining value to calculate is $N(1,1)$, but for this we need to know μ. We have

$$ \mu = \min\left\{\frac{10}{2},\ \frac{11}{\sqrt{5}},\ \frac{14}{\sqrt{8}}\right\} = \frac{11}{\sqrt{5}} \approx 4.9193, $$

and thus find that $N(1,1) = \lceil 6.9570 \rceil = 7$. Finally the scaling factor $s = N(1,0) = 5$, because we are in the B-case. As it happens, ${}_5N_2^B$ is the neighbourhood displayed in Figure 1.2(d), which was originally published by Borgefors.

Chapter 4

Construction of integer neighbourhoods: part two

In this chapter we introduce two additional classes of integer neighbourhoods, one for the B-case and one for the C-case.[1] The idea behind these neighbourhoods is a remark first made (in a slightly different setting) by Verwer in [22], that for the optimal neighbourhoods the maximum relative error is dominated by the error in the cone spanned by the vectors $(p, 0)$ and $(p, 1)$. Although this need not be true for a general approximating integer neighbourhood, it is typically so if the approximation is good.

In the terminology of Chapter 2: the maximum in equation (2.11) is usually attained by H_0. This means that we have some freedom in choosing the lengths we attribute to the vectors (p, r), where $2 \leq r \leq p$, without affecting the value of $\mathcal{E}_{\max}$. Recalling Lemma 2.4, it seems advantageous to slightly overestimate these lengths, since this will yield a better $\mathcal{E}_{\min}$, and thus also a possibly smaller $\mathcal{E}$.

This intuitive argument is merely intended as a motivation for the construction described below. Obviously the argument does not apply when $p = 1$, but we will construct neighbourhoods for that case too. Ultimately, the introduction of the new classes of integer neighbourhoods will be justified by the experimental results given in Chapter 5.

4.1 The case $p = 1$

The construction given here closely resembles the construction described in Section 3.1. We need the following auxiliary lemma.

Lemma 4.1 *Let $\frac{1}{2}\sqrt{2} < c \leq 1$ and let n be a positive integer. The following choices of n_0 and n_1 satisfy $n_0 \leq n_1 \leq 2n_0$:*

[1] In the D-case the construction below coincides with the construction in Chapter 3.

(i) $n_0 = n$, $n_1 = \lceil nc\sqrt{2}\rceil$;

(ii) $n_0 = \langle nc\rangle$, $n_1 = \lceil nc\sqrt{2}\rceil$, *unless both* $n = 2$ *and* $c < \frac{3}{4}$.

Proof. (i): It follows from $c > \frac{1}{2}\sqrt{2}$ that $\lceil nc\sqrt{2}\rceil \geq \lceil n\rceil = n$, so $n_0 \leq n_1$ is satisfied. And it follows from $c \leq 1$ that $\lceil nc\sqrt{2}\rceil \leq \lceil n\sqrt{2}\rceil \leq \lceil 2n\rceil = 2n$, so $n_1 \leq 2n_0$ also holds.

(ii): It is obvious that $n_0 \leq n_1$. First assume that $n \geq 5$. Using both that $x - \frac{1}{2} < \langle x\rangle \leq x + \frac{1}{2}$ and $x \leq \lceil x\rceil < x + 1$, we have

$$2\langle nc\rangle - \lceil nc\sqrt{2}\rceil > nc\left(2 - \sqrt{2}\right) - 2 > n\left(\sqrt{2} - 1\right) - 2 \geq 5\sqrt{2} - 7 > 0,$$

which shows that $2n_0 \geq n_1$. For $n = 4$ we have:

$$n_0 = \langle 4c\rangle = \begin{cases} 3 & \text{if } \frac{1}{2}\sqrt{2} < c < \frac{7}{8} \\ 4 & \text{if } \frac{7}{8} \leq c \leq 1 \end{cases}$$

$$n_1 = \lceil 4c\sqrt{2}\rceil = \begin{cases} 5 & \text{if } \frac{1}{2}\sqrt{2} < c \leq \frac{5}{4\sqrt{2}} \\ 6 & \text{if } \frac{5}{4\sqrt{2}} < c \leq 1 \end{cases}$$

and it is clear that $n_1 \leq 2n_0$ is always satisfied for these values. A similar argument can be used to verify the inequality for $n = 1$ and $n = 3$. For $n = 2$ however, we have:

$$n_0 = \langle 2c\rangle = \begin{cases} 1 & \text{if } \frac{1}{2}\sqrt{2} < c < \frac{3}{4} \\ 2 & \text{if } \frac{3}{4} \leq c \leq 1 \end{cases}$$

while $n_1 = 3$ for every $\frac{1}{2}\sqrt{2} < c \leq 1$. Clearly then, $n_1 > 2n_0$ if $c < \frac{3}{4}$. □

We remark that Lemma 3.1(iii) now follows as a special case of Lemma 4.1(i), by taking $c = 1$.

Definition 4.2 *Let* n *be a positive integer. The following integer neighbourhoods are defined on* M_1*:*

$$ {}^{*}_{n}N_1^B(i,j) := \begin{cases} n & \textit{if } |i| + |j| = 1 \\ \lceil n\left(1 - e_1^B\right)\sqrt{2}\rceil & \textit{if } |i| + |j| = 2 \end{cases}$$

$$ {}^{*}_{n}N_1^C(i,j) := \begin{cases} \langle n\left(1 - e_1^C\right)\rangle & \textit{if } |i| + |j| = 1 \\ \lceil n\left(1 - e_1^C\right)\sqrt{2}\rceil & \textit{if } |i| + |j| = 2 \end{cases}$$

with $e_1^B = 3 - \sqrt{2} - 2\sqrt{2 - \sqrt{2}}$ *and* $e_1^C = \frac{\sqrt{4-2\sqrt{2}}-1}{\sqrt{4-2\sqrt{2}}+1}$.

Proposition 4.3 *For $X \in \{B, C\}$, the maximum relative error of ${}^{*}_{n}N_1^X$ is*

$$ {}^{*}_{n}e_1^X = \max\left\{1 - \frac{1}{s}\min\left(n_0,\ \frac{1}{2}n_1\sqrt{2}\right),\ \frac{1}{s}\sqrt{n_0^2 + (n_1 - n_0)^2} - 1\right\}, $$

with $n_0 = {}^{*}_{n}N_1^X(1,0)$, $n_1 = {}^{*}_{n}N_1^X(1,1)$.

Proof. The result follows immediately from Lemma 4.1 and Theorem 2.2 by choosing c appropriately. □

4.2 The case $p \geq 2$

Just as ${}^{*}_{n}N_1^B$ and ${}^{*}_{n}N_1^C$ were introduced by a similar method as ${}_{n}N_1^B$ and ${}_{n}N_1^C$, the construction given here resembles the construction described in Section 3.2. We begin by proving the following lemma on reduced neighbourhoods. It uses the function $g_a(x)$ defined in Lemma 3.4.

Lemma 4.4 *Let $p \geq 2$ and $\frac{p}{\sqrt{p^2+1}} < c \leq 1$, and let n be a positive integer such that $n > \frac{5}{2cg_p(p-1)}$. The following choices of $n_0, \ldots, n_p$ satisfy $n_0 \leq n_1 \leq \ldots \leq n_p$ and $n_{j+1} + n_{j-1} \geq 2n_j$ (for $j = 1, \ldots, p-1$):*

(i) $n_0 = np$, $n_1 = \langle nc\sqrt{p^2+1}\rangle$, $n_j = \lceil nc\sqrt{p^2+j^2}\rceil$ $(j = 2, \ldots, p)$;

(ii) $n_j = \langle nc\sqrt{p^2+j^2}\rangle$ $(j = 0, 1)$, $n_j = \lceil nc\sqrt{p^2+j^2}\rceil$ $(j = 2, \ldots, p)$.

Proof. (i): It is clear that $n_1 \leq \ldots \leq n_p$, and $n_0 \leq n_1$ follows from $c > \frac{p}{\sqrt{p^2+1}}$. We have to show that $n_{j+1} + n_{j-1} \geq 2n_j$ holds for every $j = 1, \ldots, p-1$.

First take $j \geq 3$. By $x \leq \lceil x\rceil < x + 1$ we have

$$ \lceil nc\sqrt{p^2+(j+1)^2}\rceil - 2\lceil nc\sqrt{p^2+j^2}\rceil + \lceil nc\sqrt{p^2+(j-1)^2}\rceil > ncg_p(j) - 2. $$

According to Lemma 3.4 the function $g_p(j)$ is monotone decreasing, so we have

$$ ncg_p(j) - 2 \geq ncg_p(p-1) - 2 > 0, $$

by the assumption on n. The statement follows for $j \geq 3$.

Now take $j = 2$. By $x - \frac{1}{2} < \langle x\rangle \leq x + \frac{1}{2}$ and $x \leq \lceil x\rceil < x + 1$,

$$ \lceil nc\sqrt{p^2+9}\rceil - 2\lceil nc\sqrt{p^2+4}\rceil + \langle nc\sqrt{p^2+1}\rangle > ncg_p(2) - \frac{5}{2} > 0, $$

where we use again that $g_p(j)$ is monotone decreasing.

Finally, for $j = 1$ we find that

$$ \lceil nc\sqrt{p^2+4}\rceil - 2\langle nc\sqrt{p^2+1}\rangle + np > nc\left(\sqrt{p^2+4} - 2\sqrt{p^2+1}\right) + np - 1. $$

Table 4.1: Practical lower bounds on n in Lemma 4.4(i), when applied to the B-case, for $2 \leq p \leq 10$. The values of n are indicated for which choosing $n_0 = np$, $n_1 = \langle n(1-e_p^B)\sqrt{p^2+1}\rangle$, $n_j = \lceil n(1-e_p^B)\sqrt{p^2+j^2}\rceil$ $(j = 2,\ldots,p)$ gives a reduced neighbourhood that satisfies (2.7).

p	OK for
2	$n \geq 1$
3	$n = 1$ and for $n \geq 3$
4	$n = 2$ and for $n \geq 4$
5	$n = 2, 4$ and for $n \geq 7$
6	$n = 2, 5, 6$ and for $n \geq 8$
7	$n = 2, 6, 7, 9$ and for $n \geq 11$
8	$n = 6, 7, 8, 11, 12$ and for $n \geq 14$
9	$n = 3, 7, 8, 9$ and for $n \geq 11$
10	$n = 3, 8, 10$ and for $n \geq 12$

By $c \leq 1$ this is larger than or equal to $ncg_p(1) - 1$, which is positive.

(ii): It is easy to see that $n_0 \leq n_1 \leq \ldots \leq n_p$. Moreover, the proof of $n_{j+1} + n_{j-1} \geq 2n_j$ for $j = 1,\ldots,p-1$ is analogous to the proof given under (i). (In fact, the proof for $j \geq 2$ can be used verbatim.) □

The next proposition is an analogue to Proposition 3.6. Lemma 4.4 provides a lower bound on n, which can be sharpened by systematically testing all smaller values of n. The results of this exercise are displayed in Tables 4.1 and 4.2.

Proposition 4.5 *Let $2 \leq p \leq 10$ and n a positive integer. The following reduced neighbourhoods on M_p satisfy $n_{j+1} + n_{j-1} \geq 2n_j$ (for $j = 1,\ldots,p-1$):*

$$ {}^*_n\mathcal{N}_p^B(i,j) := \begin{cases} np & \text{if } |i| = p \text{ and } j = 0 \\ \langle n\left(1-e_p^B\right)\sqrt{p^2+1}\rangle & \text{if } |i| = p \text{ and } |j| = 1 \\ \lceil n\left(1-e_p^B\right)\sqrt{p^2+j^2}\rceil & \text{if } |i| = p \text{ and } 2 \leq |j| \leq p \\ \infty & \text{elsewhere} \end{cases} $$

for the values of n displayed in Table 4.1;

$$ {}^*_n\mathcal{N}_p^C(i,j) := \begin{cases} \langle n\left(1-e_p^C\right)\sqrt{p^2+j^2}\rangle & \text{if } |i| = p \text{ and } |j| \leq 1 \\ \lceil n\left(1-e_p^C\right)\sqrt{p^2+j^2}\rceil & \text{if } |i| = p \text{ and } 2 \leq |j| \leq p \\ \infty & \text{elsewhere} \end{cases} $$

for the values of n displayed in Table 4.2.

Table 4.2: Practical lower bounds on n in Lemma 4.4(ii), when applied to the C-case, for $2 \leq p \leq 10$. The values of n are indicated for which choosing $n_j = \langle n(1-e_p^C)\sqrt{p^2+j^2}\rangle$ $(j = 0, 1)$, $n_j = \lceil n(1-e_p^C)\sqrt{p^2+j^2}\rceil$ $(j = 2, \ldots, p)$ gives a reduced neighbourhood that satisfies (2.7).

p	OK for
2	$n \geq 1$
3	$n = 1$ and for $n \geq 3$
4	$n = 2$ and for $n \geq 4$
5	$n = 2, 4$ and for $n \geq 6$
6	$n = 2$ and for $n \geq 5$
7	$n = 2, 6, 7, 9$ and for $n \geq 11$
8	$n = 6, 7, 8, 10, 11, 12$ and for $n \geq 14$
9	$n = 3, 8, 9$ and for $n \geq 11$
10	$n = 3, 8, 9, 10$ and for $n \geq 12$

In order to use Proposition 2.5 to calculate the maximum relative errors of these reduced neighbourhoods, we must check that condition (2.10) holds. Maple software was used for parts of the proof of the following lemma.

Lemma 4.6 *Let $p \geq 2$. For every $n \geq 1$ the neighbourhoods ${}^*_n\mathcal{N}_p^B$ and ${}^*_n\mathcal{N}_p^C$ defined in Proposition 4.5 satisfy $(r+1)n_r > rn_{r+1}$ (for $r = 0, \ldots, p-1$).*

Proof. The proof is given in Appendix A.5. □

We now extend these reduced neighbourhoods to full integer neighbourhoods in the same way as before.

Definition 4.7 *Let $2 \leq p \leq 10$ and n a positive integer. The following neighbourhoods are defined on M_p:*

$$
{}^*_nN_p^B(i,j) := \begin{cases} np & \text{if } |i| = p \text{ and } j = 0 \\ \langle n\left(1-e_p^B\right)\sqrt{p^2+1}\rangle & \text{if } |i| = p \text{ and } |j| = 1 \\ \lceil n\left(1-e_p^B\right)\sqrt{p^2+j^2}\rceil & \text{if } |i| = p \text{ and } 2 \leq |j| \leq p \\ \lceil \mu\sqrt{i^2+j^2}\rceil & \text{for all other vectors in } M_p^* \end{cases}
$$

where $\mu = \min\limits_{0\leq k\leq p} \frac{{}^*_nN_p^B(p,k)}{\sqrt{p^2+k^2}}$, for the values of n displayed in Table 4.1;

$$
{}^*_nN_p^C(i,j) := \begin{cases} \langle n\left(1-e_p^C\right)\sqrt{p^2+j^2}\rangle & \text{if } |i| = p \text{ and } |j| \leq 1 \\ \lceil n\left(1-e_p^C\right)\sqrt{p^2+j^2}\rceil & \text{if } |i| = p \text{ and } 2 \leq |j| \leq p \\ \lceil \mu\sqrt{i^2+j^2}\rceil & \text{for all other vectors in } M_p^* \end{cases}
$$

where $\mu = \min_{0 \le k \le p} \frac{{}^*_n N^C_p(p,k)}{\sqrt{p^2+k^2}}$, *for the values of* n *displayed in Table* 4.2.
The values of e^B_p *and* e^C_p *are given by* (1.10) *and* (1.12).

Proposition 4.8 *For* $X \in \{B, C\}$, *the maximum relative error of* ${}^*_n N^X_p$ *satisfies*

$$ {}^*_n e^X_p \le \max \left\{ 1 - \frac{1}{s} \min_{0 \le k \le p} \frac{{}^*_n N^X_p(p,k)}{\sqrt{p^2+k^2}},\ \frac{1}{s} \max_{0 \le r \le p-1} H_r - 1 \right\} $$

where, putting $n_j = {}^*_n N^X_p(p,j)$,

$$ H_r = \begin{cases} \sqrt{\frac{1}{p^2}\left\{(r+1)n_r - r n_{r+1}\right\}^2 + (n_{r+1}-n_r)^2} & \text{if } \lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - r n_{r+1}} \rfloor = r \\ \frac{n_r}{\sqrt{p^2+r^2}} & \text{if } \lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - r n_{r+1}} \rfloor < r \\ \frac{n_{r+1}}{\sqrt{p^2+(r+1)^2}} & \text{if } \lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - r n_{r+1}} \rfloor > r \end{cases} $$

Proof. Completely analogous to the proof of Proposition 3.9. □

Chapter 5

Results

In the previous chapters five classes of integer neighbourhoods were defined: ${}_nN_p^B$, ${}_n^*N_p^B$, ${}_nN_p^C$, ${}_n^*N_p^C$ and ${}_nN_p^D$. These neighbourhoods yield distance transformations that approximate the optimal distance transformations w_p^B, w_p^C and w_p^D given in Section 1.4. We also worked out a strategy for determining the maximum relative errors of these neighbourhoods.

For a fixed p the neighbourhoods in each of the classes mentioned above are completely determined by one parameter, viz. n. (In the C-case the distance transformation also depends on the choice of scaling factor s. However we always take the optimal s, defined in expression (2.15), which is indirectly determined by n.) Broadly speaking, distance transformations with larger values of n will have a maximum relative error closer to the optimal value, since the elements of the optimal neighbourhood are approximated better by fractions with higher denominators. But certain values of n yield better results than other values of the same magnitude, and simply increasing the denominator does not guarantee us that the maximum relative error will be reduced. As an example, for $p = 2$, ${}_5N_2^B$ corresponds to Figure 1.2(d). It outperforms every neighbourhood from this class up to ${}_{31}N_2^B$.

Let N be an integer neighbourhood from one of the five classes defined above, and let its parameter value be k. We call N a *best* neighbourhood if the associated maximum relative error minimises the maximum relative error of all neighbourhoods with the same superscript (i.e. either B, C or D) with $1 \leq n \leq k$. This is the type of neighbourhood we are interested in.

In the following sections the best choices of n will be derived for each case, for $1 \leq p \leq 10$ and $n \leq 1000$.

5.1 The B-case

Appendices B.1 and B.2 contain programs that can be used to calculate the elements of neighbourhoods of the form ${}_nN_p^B$ and ${}_n^*N_p^B$, respectively. The program given in Appendix B.6 implements the theory of Chapter 2 to calculate the maximum relative errors of these neighbourhoods (or rather, to calculate the upper bound on the maximum relative error derived in Propositions 3.9 and 4.8). These programs were used to generate a list of all best neighbourhoods, for a given $p \geq 2$, with parameter value $n \leq 1000$. The same result was obtained for $p = 1$, using similar programs.

Not every best neighbourhood is particularly interesting or useful. Especially for higher values of p, many best neighbourhoods occur that barely improve on the previous best maximum relative error, and such neighbourhoods have been omitted.[1] We will therefore introduce a measure of quality for best neighbourhoods.

For a given neighbourhood ${}_nN_p^B$ with maximum relative error ${}_ne_p^B$, we denote by $\delta(n,p)$ the distance to the optimal error e_p^B:

$$\delta(n,p) := {}_ne_p^B - e_p^B. \tag{5.1}$$

We make the same definition for neighbourhoods of the form ${}_n^*N_p^B$.

Obviously, a neighbourhood will be particularly of interest if both n and $\delta(n,p)$ are relatively small. In that case, the product $\delta(n,p) \cdot n^c$, where $c \geq 1$ is some constant, will also be relatively small. For convenience we take the negative logarithm of this expression as a measure of quality:

$$q(n,p) := -{}^{10}\log \delta(n,p) - c \cdot {}^{10}\log n. \tag{5.2}$$

Note that the interesting neighbourhoods will have high values of $q(n,p)$. The constant c is chosen such that a priori $q(n,p)$ is not biased towards a particular value of n. In the B-case we used the value $c = 1.9$.

We list all best neighbourhoods here that satisfy the criterion:

$$q(n,p) \geq Q, \tag{5.3}$$

where $Q = 0.9$ in this case. Since both c and Q are chosen to suit the experimental data, there is no rigour behind this criterion, and there is no reason to apply it rigourously. Accordingly, we will allow for exceptions:

- For each p small values of n are discarded if the maximum relative errors do not improve on the theoretical optima of smaller masks.
- On the other hand, the first value of n that *does* improve on the previous best theoretical optimum is listed, regardless of its $q(n,p)$-value.
- For each p the best neighbourhood with the lowest maximum relative error (i.e. the last best neighbourhood found with $n \leq 1000$) is always printed.

[1] A complete list for each case is provided online, however, at: `http://www.math.leidenuniv.nl/~scholtus/chamfer.htm`.

Table 5.1: A selection of best integer neighbourhoods for the B-case. Plain values of n refer to the class ${}_nN_p^B$ and starred values refer to ${}_n^*N_p^B$.

n	max.rel.err.	$q(n,p)$	n	max.rel.err.	$q(n,p)$
$p=1$		$e_1^B \approx 0.05505271$	*415	0.00335198	1.00
1	0.29289322	0.62	*476	0.00335099	2.01
3	0.05719096	1.76	$p=6$		$e_6^B \approx 0.00234378$
110	0.05505474	1.81	*25	0.00319490	0.41
993	0.05505468	0.01	*43	0.00243079	0.96
$p=2$		$e_2^B \approx 0.01869475$	*44	0.00239705	1.15
4	0.03077641	0.77	*58	0.00237530	1.15
5	0.01980390	1.63	*73	0.00234587	2.14
*31	0.01901534	0.66	*321	0.00234583	0.93
36	0.01872893	1.51	*686	0.00234428	0.91
139	0.01870398	0.96	*759	0.00234414	0.97
175	0.01869865	1.15	*832	0.00234404	1.04
314	0.01869518	1.62	*905	0.00234395	1.15
$p=3$		$e_3^B \approx 0.00893928$	*978	0.00234387	1.36
8	0.01178823	0.83	$p=7$		$e_7^B \approx 0.00172949$
*15	0.00915300	1.44	18	0.00219376	0.95
*37	0.00908944	0.84	*33	0.00183486	1.09
52	0.00901997	0.83	*51	0.00173160	2.43
*67	0.00898172	0.90	$p=8$		$e_8^B \approx 0.00132791$
*82	0.00895750	1.10	*37	0.00145985	0.90
97	0.00894079	2.05	*58	0.00133680	1.70
*791	0.00893978	0.79	*97	0.00132858	2.40
*888	0.00893933	1.70	*640	0.00132846	0.93
$p=4$		$e_4^B \approx 0.00516800$	*737	0.00132835	0.91
9	0.00619201	1.18	*931	0.00132820	0.90
19	0.00552490	1.02	$p=9$		$e_9^B \approx 0.00105127$
*29	0.00533653	0.99	*60	0.00126463	0.29
*39	0.00524594	1.09	*65	0.00106452	1.43
49	0.00519268	1.40	*87	0.00105638	1.61
*108	0.00517352	1.39	*109	0.00105155	2.68
167	0.00516824	2.40	$p=10$		$e_{10}^B \approx 0.00085272$
$p=5$		$e_5^B \approx 0.00335091$	*67	0.00100195	0.36
13	0.00460478	0.79	*73	0.00087362	1.14
*24	0.00360461	0.97	*97	0.00085956	1.39
*37	0.00351802	0.80	*121	0.00085340	2.21
*49	0.00340984	1.02	*460	0.00085287	1.76
*61	0.00335369	2.16	*581	0.00085282	1.75

- Neighbourhoods with $q(n,p)$ smaller than Q may be printed, if we feel this is useful.

Table 5.1 shows a selection of best neighbourhoods we found in the B-case, for $1 \leq p \leq 10$ and $n \leq 1000$.

Some of these integer neighbourhoods have been published previously. The traditional chessboard distance transformation (Figure 1.2(b)) is given by ${}_1N_1^B$. Borgefors suggested three neighbourhoods in [3] that correspond to ${}_3N_1^B$, ${}_5N_2^B$ and ${}_{12}N_3^B$. The first two of these are very good neighbourhoods (in the sense that their $q(n,p)$-values are uncommonly high: 1.76 and 1.63, respectively), but the third one is surpassed in maximum relative error by ${}_8N_3^B$. Coquin and Bolon suggested ${}_{67}^{*}N_3^B$ in [8]. This neighbourhood is also listed in Table 5.1.

Integer neighbourhoods have also been published that do not fall within one of the classes ${}_nN_p^B$ and ${}_n^{*}N_p^B$. However, we did not find examples in the literature that achieve a better maximum relative error than the values given in Table 5.1.

5.2 The D-case

Appendix B.5 contains a program which calculates the elements of neighbourhoods of the form ${}_nN_p^D$. Just as in the B-case, we calculate the maximum relative errors to find the best neighbourhoods of this class.

Again we need a criterion to decide which of the best neighbourhoods are listed here. For a given neighbourhood ${}_nN_p^D$ with maximum relative error ${}_ne_p^D$, the distance to the optimal maximum relative error e_p^D is denoted by $\delta(n,p)$; that is to say, we take definition (5.1) and replace the superscript B by D. As a measure of quality we take $q(n,p)$ as defined in (5.2), this time with $c = 1.8$. A neighbourhood is listed if $q(n,p) \geq Q$, where $Q = 0.6$ in this case. (The same reservations hold as in the B-case.)

Table 5.2 lists some best neighbourhoods we found in the D-case, for $1 \leq p \leq 10$ and $n \leq 1000$.

We did not find any previously published integer neighbourhoods for the D-case, other than the classical city block distance transformation (Figure 1.2(a)), which corresponds to ${}_1N_1^D$ (listed in Table 5.2).

5.3 The C-case

Appendices B.3 and B.4 contain programs that can be used to calculate the elements of a neighbourhood from the classes ${}_nN_p^C$ and ${}_n^{*}N_p^C$. As before we compute the maximum relative errors for $n \leq 1000$, but this time the scaling factor s has to be calculated also, from

$$s = \frac{1}{2}\left(c_{\max} + c_{\min}\right).$$

Table 5.2: A selection of best integer neighbourhoods from the class ${}_nN_p^D$.

n	max.rel.err.	$q(n,p)$	n	max.rel.err.	$q(n,p)$
$p=1$		$e_1^D \approx 0.08239220$	313	0.00489264	1.17
1	0.41421356	0.48	414	0.00489188	1.14
2	0.11803399	0.91	515	0.00489142	1.14
7	0.08796759	0.73	616	0.00489111	1.17
12	0.08333333	1.08	717	0.00489089	1.24
41	0.08255322	0.89	818	0.00489072	1.36
70	0.08241981	1.24	919	0.00489059	1.59
239	0.08239694	1.04	$p=6$		$e_6^D \approx 0.00341897$
408	0.08239301	1.39	33	0.00469513	0.16
$p=2$		$e_2^D \approx 0.02748630$	46	0.00377360	0.46
3	0.06718737	0.54	157	0.00342227	1.53
8	0.03077641	0.86	302	0.00342053	1.34
21	0.02795396	0.95	447	0.00341992	1.25
38	0.02766443	0.91	592	0.00341961	1.20
55	0.02755427	1.04	737	0.00341942	1.19
72	0.02749621	1.66	882	0.00341930	1.18
377	0.02748774	1.20	$p=7$		$e_7^D \approx 0.00252214$
682	0.02748685	1.16	42	0.00295736	0.44
987	0.02748651	1.29	139	0.00258452	0.35
$p=3$		$e_3^D \approx 0.01308146$	182	0.00254777	0.52
6	0.02439383	0.55	239	0.00252653	1.08
11	0.01639454	0.61	408	0.00252289	1.43
18	0.01379376	0.89	605	0.00252260	1.33
43	0.01316376	1.14	802	0.00252246	1.27
80	0.01311710	1.02	999	0.00252237	1.24
117	0.01309997	1.01	$p=8$		$e_8^D \approx 0.00193614$
154	0.01309107	1.08	60	0.00236050	0.17
191	0.01308562	1.28	64	0.00207305	0.61
228	0.01308194	2.07	272	0.00195122	0.44
$p=4$		$e_4^D \approx 0.00754900$	337	0.00193967	0.90
14	0.01157207	0.33	530	0.00193654	1.49
40	0.00778222	0.75	787	0.00193639	1.39
73	0.00757126	1.30	$p=9$		$e_9^D \approx 0.00153258$
138	0.00755912	1.14	53	0.00191660	0.31
203	0.00755476	1.09	90	0.00162496	0.52
268	0.00755251	1.08	378	0.00154202	0.39
333	0.00755114	1.13	469	0.00153546	0.73
398	0.00755022	1.23	668	0.00153281	1.55
463	0.00754956	1.45	993	0.00153272	1.46
528	0.00754906	2.32	$p=10$		$e_{10}^D \approx 0.00124302$
$p=5$		$e_5^D \approx 0.00480947$	110	0.00152776	-0.13
27	0.00751633	0.00	120	0.00129916	0.51
70	0.00498756	0.69	520	0.00124922	0.32
111	0.00489832	1.42	621	0.00124520	0.63
212	0.00489413	1.25	822	0.00124315	1.64

In Table 5.3 best neighbourhoods for the C-case are listed for $1 \leq p \leq 10$ and $n \leq 1000$. Just as before, the less interesting values of n have been filtered out by applying the criterion $q(n,p) \geq Q$. This time we put $c = 2$ for $p \in \{1, 2\}$ and $c = 2.1$ for $p \geq 3$, and $Q = 0.8$.

Some of these neighbourhoods have been published before. In [22] Verwer suggested ${}_2N_1^C$, ${}_5N_1^C$, ${}_{12}^{\ *}N_1^C$ and ${}_5N_2^C$, all of which are listed in Table 5.3. He also suggested ${}_4^*N_2^C$ and ${}_9^*N_2^C$, but a lower maximum relative error is achieved by ${}_4N_2^C$ and ${}_8^*N_2^C$, respectively. Finally, he suggested a neighbourhood with $s = 17.2174$ that does not correspond to either ${}_{17}N_2^C$ or ${}_{17}^{\ *}N_2^C$, but ${}_{13}^{\ *}N_2^C$ already yields a lower maximum relative error.

Coquin and Bolon suggested ${}_{25}N_1^C$ in [8], but ${}_{12}^{\ *}N_1^C$ achieves the same maximum relative error. In [6] Butt and Maragos suggested ${}_{73}N_1^C$ (listed in Table 5.3), but with a different (non-optimal) scaling factor. Finally in [18] Thiel suggested ${}_{73}N_2^C$, which is listed in Table 5.3.

Just as for the B-case, integer neighbourhoods have also been published that do not fall within one of the classes derived here.[2] However, we did not find any examples in the literature that achieve a better maximum relative error than the values given in Table 5.3.

5.4 Concluding remarks

The tables given in the previous sections make it possible to select mask size p and parameter n to guarantee that the approximation of the Euclidean distance does not exceed a prescribed maximum relative error. In each case the tables run down to an error of about 0.1%.

Of course, the use of a neighbourhood with a larger value of n requires more computer memory. The following argument[3] makes this more precise. Suppose N is an integer neighbourhood on M_p from one of the classes from Chapters 3 and 4, with parameter value n. If we want to use N to get a distance transformation of a digitised picture of size $D \times D$, the largest possible distance to be approximated is the length of the diagonal from the lower left-hand corner to the upper right-hand corner. This distance is approximated (before the scaling factor is divided out) by

$$\frac{D}{p} N(p,p) \approx \frac{D}{p} np\sqrt{2} = nD\sqrt{2}.$$

Observe that this is independent of the mask-size p. If we are using i bits to code this distance, it should clearly hold that

$$2^i > nD\sqrt{2}. \tag{5.4}$$

[2]For instance, Vossepoel (cf. [23]) and Borgefors (cf. [5]) give many examples for the C-case. Since they optimised the maximum absolute error (equation (1.8)), their scaling factors are not optimal from our point of view.

[3]This is based on a similar argument used by Coquin and Bolon in [8].

Table 5.3: A selection of best integer neighbourhoods for the C-case. Plain values of n refer to the class ${}_nN_p^C$ and starred values refer to ${}_n^*N_p^C$.

n	s	max.rel.err.	$q(n,p)$
$p=1$			$e_1 \approx 0.03956613$
1	0.85355339	0.17157288	0.88
2	2.11803399	0.05572809	1.19
5	5.16745614	0.04213072	1.19
*12	12.5000000	0.04000000	1.20
30	30.1880438	0.03964039	1.18
73	72.8846935	0.03957887	1.17
176	175.961001	0.03956831	1.17
425	424.808175	0.03956650	1.18
$p=2$			$e_2 \approx 0.01355683$
*3	3.10078106	0.03250183	0.77
5	5.00918453	0.01793405	0.96
*8	8.12310563	0.01515500	0.99
*13	13.1554292	0.01415650	0.99
26	25.8437744	0.01364350	1.23
*47	47.1416827	0.01361179	0.92
73	72.9898635	0.01356166	1.59
*309	309.191412	0.01355710	1.59
*846	845.968804	0.01355701	0.89
$p=3$			$e_3 \approx 0.00649823$
*6	6.07318149	0.01204994	0.62
13	13.0582854	0.00724912	0.79
31	30.8691343	0.00655890	1.09
*62	62.0683152	0.00650736	1.28
*217	217.077453	0.00649900	1.21
242	241.905114	0.00649847	1.61
701	700.887913	0.00649830	1.18
$p=4$			$e_4 \approx 0.00376031$
9	8.99982852	0.00617307	0.61
17	17.0439028	0.00430301	0.68
49	48.9320625	0.00379670	0.89
*57	56.9650442	0.00377502	1.15
106	105.898965	0.00376741	0.90
*212	212.046069	0.00376295	0.69
261	260.980010	0.00376205	0.68
367	366.879879	0.00376112	0.71
*530	529.992951	0.00376033	1.98
$p=5$			$e_5 \approx 0.00243927$
*20	20.0574267	0.00334964	0.31
*31	31.0580500	0.00279390	0.32
70	69.9750244	0.00250233	0.33
*101	101.047409	0.00244844	0.83
152	151.969025	0.00244295	0.85
*162	161.994921	0.00243974	1.69
*820	820.000304	0.00243939	0.80
*901	900.997841	0.00243934	0.95
*982	981.995378	0.00243930	1.24
$p=6$			$e_6 \approx 0.00170657$
*24	24.0415991	0.00211365	0.49
62	61.9245898	0.00178557	0.34
73	72.9566837	0.00175354	0.42
157	156.935885	0.00171547	0.44
351	350.932451	0.00170722	0.84
*460	459.951384	0.00170673	1.20
932	931.923860	0.00170671	0.62
$p=7$			$e_7 \approx 0.00125948$
*44	44.0287673	0.00166679	-0.06
71	70.9478166	0.00134912	0.16
99	98.9805381	0.00128318	0.43
*225	224.998732	0.00126421	0.39
296	295.943568	0.00126015	0.98
*310	309.961759	0.00125954	1.99
$p=8$			$e_8 \approx 0.00096713$
*52	52.0300425	0.00122570	-0.02
81	80.9500295	0.00098064	0.86
97	96.9658782	0.00097715	0.83
*256	256.000000	0.00097656	-0.03
*386	385.997554	0.00096779	0.75
*402	402.013404	0.00096746	1.01
*611	610.965832	0.00096716	1.67
$p=9$			$e_9 \approx 0.00076570$
82	81.9651408	0.00093030	-0.24
90	89.9658728	0.00085570	-0.06
109	108.967607	0.00079347	0.28
127	126.983824	0.00077176	0.80
524	523.956829	0.00076585	1.11
$p=10$			$e_{10} \approx 0.00062112$
*66	66.0206535	0.00075261	0.06
100	99.9644855	0.00066583	0.15
*161	160.998136	0.00062499	0.78
581	580.961900	0.00062293	-0.06
742	741.961277	0.00062170	0.21
*943	942.985501	0.00062120	0.85
*963	962.998117	0.00062113	1.73

In particular, this shows that for $i = 32$ all neighbourhoods with $n \leq 1000$ can be used for distance transformations with pictures of sizes up to $D = \frac{2^{32}}{1000\sqrt{2}} \approx 3 \times 10^6$.

Appendix A

Mathematical bits and pieces

A.1 Butt and Maragos revisited

In this section we give a proof of the fact mentioned in Section 1.4, that the optimal error found by Butt and Maragos in [6] is equal to the one found by Verwer in [22], although their definitions of error are different (expressions (1.7) and (1.9)). We are talking about the C-case here, as that is the only case treated by Butt and Maragos (and Verwer). We will use the notation used by Butt and Maragos in their paper, and introduce without proof the more peripheral steps of their approach. The reader is referred to [6] for full details.

To make the analysis easier, the weighted distance function is extended to the real plane. (This corresponds to taking the limit in (1.9).) For $r > 0$, the *chamfer polygon*[1] of size r consists of all points on the plane for which the weighted distance to the origin is at most equal to r. As usual we can restrict the analysis to the first octant. A point on the edge of the chamfer polygon has weighted distance r, and its Euclidean distance is given by a function $L(\theta)$, where θ is the angle between the horizontal axis and the vector describing the point. The error at this point is defined as

$$E(\theta) := \frac{r - L(\theta)}{r}, \tag{A.1}$$

and the overall error of the weighted distance is the maximum of $|E(\theta)|$. In our own notation, this is e_{BM} as defined in (1.9).

Consider a neighbourhood w defined on M_p, with $p \geq 1$. Assuming that the neighbourhood values are reasonably well behaved (and in the optimal case, they will be), the error of the weighted distance is dominated by the error on the first sector of the chamfer polygon. This sector (OAZ in

[1]This term is used by Butt and Maragos. We recall that weighted distance transformations are sometimes referred to as *chamfer distances*.

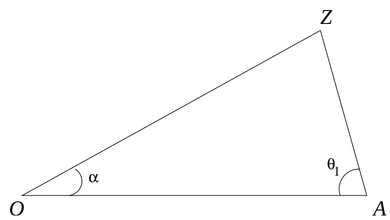

Figure A.1: The first sector of the chamfer polygon. The distance OA equals r.

Figure A.1) consists of all points for which $0 \leq \theta \leq \alpha$, where

$$\alpha := \arctan\left(\frac{1}{p}\right). \tag{A.2}$$

Furthermore in this sector $L(\theta)$ depends only on the values $a := w(1,0)$ and $z := w(p,1)$. More precisely, Butt and Maragos have established that the Euclidean distance in this first sector is

$$L(\theta) := \frac{r\sin(\theta_1)}{a\sin(\theta_1+\theta)} \qquad (0 \leq \theta \leq \alpha), \tag{A.3}$$

where

$$\theta_1 := \arctan\left(\frac{a}{z-pa}\right).$$

They also argue that in the optimal case we must have $z = a\sqrt{p^2+1}$, so θ_1 reduces to

$$\theta_1 = \arctan\left(\frac{1}{\sqrt{p^2+1}-p}\right). \tag{A.4}$$

It is clear from (A.1) that $E(\theta)$ achieves its maximum when $L(\theta)$ achieves its minimum. It is easy to see that this happens for $\bar{\theta} = 90^\circ - \theta_1$, with $L(\bar{\theta}) = \frac{r\sin(\theta_1)}{a}$. The optimal value of a can now be obtained by solving

$$E(\bar{\theta}) = -E(0) = -E(\alpha). \tag{A.5}$$

We will use the following properties, the first of which is trivial.

Property A.1 $\sin\left(\arctan\frac{y}{x}\right) = \frac{y}{\sqrt{x^2+y^2}}$.

Property A.2 $\alpha = 2\bar{\theta}$.

Proof. In the big triangle shown in Figure A.2 we have $\alpha = 180^\circ - \theta_1 - (\bar{\theta}+\beta)$. But this triangle is isosceles, so $\bar{\theta}+\beta = \theta_1$, which in turn gives $\alpha = 2\,(90^\circ - \theta_1) = 2\bar{\theta}$. □

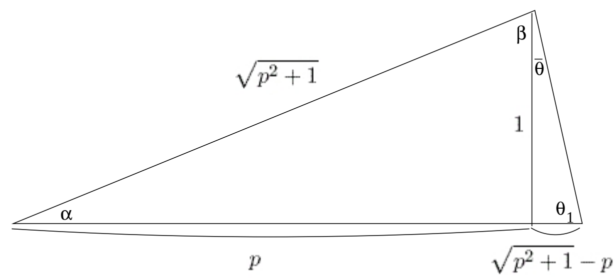

Figure A.2: An isosceles triangle used in the proof of Property A.2.

It follows easily from Property A.2 that $E(0) = E(\alpha)$, for we have (using that $\bar{\theta} = 90^\circ - \theta_1$):

$$L(\alpha) = \frac{r\sin(\theta_1)}{a\sin(\theta_1 + 2\bar{\theta})} = \frac{r\sin(\theta_1)}{a\sin(90^\circ + \bar{\theta})} = \frac{r\sin(\theta_1)}{a\sin(90^\circ - \bar{\theta})} = \frac{r}{a} = L(0).$$

Thus it is sufficient to solve $E(\bar{\theta}) = -E(0)$, that is:

$$1 - \frac{\sin(\theta_1)}{a} = \frac{1}{a} - 1.$$

The solution is

$$a = \frac{1 + \sin(\theta_1)}{2},$$

and the resulting value of $E(\bar{\theta})$ (and thus of $\max_\theta |E(\theta)|$) is:

$$E(\bar{\theta}) = 1 - \frac{2\sin(\theta_1)}{1 + \sin(\theta_1)} = 1 - \frac{2}{\frac{1}{\sin(\theta_1)} + 1}.$$

Using Property A.1 with (A.4), this becomes:

$$E(\bar{\theta}) = 1 - \frac{2}{\sqrt{\left(\sqrt{p^2+1} - p\right)^2 + 1} + 1} = \frac{\sqrt{2p^2 + 2 - 2p\sqrt{p^2+1}} - 1}{\sqrt{2p^2 + 2 - 2p\sqrt{p^2+1}} + 1}.$$

Comparing with (1.12), we see that this is exactly e_p^C.

A.2 Derivation of equation (2.11)

The function h_r is defined, for $r = 0, 1, \ldots, p-1$, as

$$h_r(t) = \frac{1}{s}\frac{\frac{1}{p}\{(r+1)n_r - rn_{r+1}\} + (n_{r+1} - n_r)\,t}{\sqrt{1+t^2}} \qquad \left(\frac{r}{p} \le t \le \frac{r+1}{p}\right).$$

Our objective is to determine

$$\max_{0 \le r < p} \max_{\frac{r}{p} \le t \le \frac{r+1}{p}} h_r(t).$$

We begin by finding the maximum of h_r on the whole of $\mathbb{R}$. The derivative of h_r is

$$h_r'(t) = \frac{1}{s} \frac{n_{r+1} - n_r - \frac{1}{p}\left\{(r+1)n_r - rn_{r+1}\right\} t}{(1+t^2)^{3/2}}.$$

The equation $h_r'(t) = 0$ has only one solution, $\bar{t} = \frac{p(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}}$, and it follows from $(r+1)n_r > rn_{r+1}$ that $h_r(\bar{t})$ is indeed the maximum of h_r.

If $\bar{t}$ happens to satisfy $\frac{r}{p} \le \bar{t} \le \frac{r+1}{p}$, then h_r attains its maximum on $\left[\frac{r}{p}, \frac{r+1}{p}\right]$ at $\bar{t}$. If $\bar{t} < \frac{r}{p}$, then the maximum on $\left[\frac{r}{p}, \frac{r+1}{p}\right]$ is attained at $\frac{r}{p}$, and if $\bar{t} > \frac{r+1}{p}$, the maximum on $\left[\frac{r}{p}, \frac{r+1}{p}\right]$ is attained at $\frac{r+1}{p}$.

The resulting values of $h_r(t)$ are

$$h_r\left(\bar{t}\right) = \frac{1}{s}\sqrt{\frac{1}{p^2}\left\{(r+1)n_r - rn_{r+1}\right\}^2 + (n_{r+1}-n_r)^2}$$

and

$$h_r\left(\frac{r}{p}\right) = \frac{1}{s}\frac{n_r}{\sqrt{p^2+r^2}}, \; h_r\left(\frac{r+1}{p}\right) = \frac{1}{s}\frac{n_{r+1}}{\sqrt{p^2+(r+1)^2}},$$

respectively. Writing $\frac{1}{s}H_r = \max_{\frac{r}{p} \le t \le \frac{r+1}{p}} h_r(t)$, we conclude that

$$H_r = \begin{cases} \sqrt{\frac{1}{p^2}\left\{(r+1)n_r - rn_{r+1}\right\}^2 + (n_{r+1}-n_r)^2} & \text{if } \left\lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \right\rfloor = r \\ \frac{n_r}{\sqrt{p^2+r^2}} & \text{if } \left\lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \right\rfloor < r \\ \frac{n_{r+1}}{\sqrt{p^2+(r+1)^2}} & \text{if } \left\lfloor \frac{p^2(n_{r+1}-n_r)}{(r+1)n_r - rn_{r+1}} \right\rfloor > r \end{cases}$$

Equation (2.11) now follows.

A.3 The maximum relative error of optimal neighbourhoods

In this section we use Theorem 2.2 and Theorem 2.7 to bound the maximum relative error of the optimal neighbourhoods w_p^B, w_p^C and w_p^D. The resulting values turn out to be the optimal errors e_p^B, e_p^C and e_p^D, respectively. Therefore our expressions for the maximum relative error are consistent with the theoretical results of Hajdu, Hajdu and Tijdeman in [11]. In particular, this establishes that equality holds in (2.13) for the optimal neighbourhoods.

We begin with the case $p = 1$. Theorem 2.2 is formulated to suit integer neighbourhoods, but as we did not use the fact that the elements of N are integers anywhere in our derivation, it is clear that the theorem may be adapted to general neighbourhoods as follows:

Theorem A.3 *Let w be a neighbourhood on M_1 with $w(1,0) = w_0$ and $w(1,1) = w_1$, where $w_0 \leq w_1 \leq 2w_0$. The maximum relative error of w is given by*

$$e = \max \left\{ 1 - \min\left(w_0,\ \frac{1}{2} w_1 \sqrt{2} \right),\ \sqrt{w_1^2 - 2w_0 w_1 + 2w_0^2} - 1 \right\}.$$

The optimal neighbourhood w_1^B is defined in (1.13) and has $w_1^B(1,0) = 1$, $w_1^B(1,1) = (1 - e_1^B)\sqrt{2}$. By (1.10) the optimal error e_1^B equals $3 - \sqrt{2} - 2\sqrt{2 - \sqrt{2}}$. It follows immediately that

$$1 - \min\left(w_1^B(1,0),\ \frac{1}{2} w_1^B(1,1)\sqrt{2} \right) = e_1^B.$$

It is not difficult (but it takes some time) to verify the identity

$$2(1 - e_1^B)^2 - 2(1 - e_1^B)\sqrt{2} + 2 = (1 + e_1^B)^2,$$

from which it follows that

$$\sqrt{\left(w_1^B(1,1)\right)^2 - 2\left(w_1^B(1,0)\right)\left(w_1^B(1,1)\right) + 2\left(w_1^B(1,0)\right)^2} - 1 = e_1^B.$$

Thus by Theorem A.3 the maximum relative error of w_1^B is indeed e_1^B.

Similar arguments can be used to verify that Theorem A.3 yields e_1^C as the maximum relative error of w_1^C, and e_1^D for w_1^D. Of course this comes as no surprise: it would have been a serious problem if Theorem A.3 yielded different values.

It is a different matter for $p \geq 2$ however, for in this case we only have an upper bound on the maximum relative error. We have the following adapted version of Theorem 2.7 (using the formulation of Corollary 2.8):

Theorem A.4 *Let $p \geq 2$ and let w be a neighbourhood on M_p. Write $w_j = w(p,j)$ for $j = 0, \ldots, p$. If the inequalities*

(i) $w_0 \leq w_1 \leq \ldots \leq w_p$,

(ii) $w_{j+1} + w_{j-1} \geq 2w_j$ *(for $j = 1, \ldots, p-1$)*,

(iii) $(r+1)w_r > r w_{r+1}$ *(for $r = 0, \ldots, p-1$)*,

(iv) $1 + \frac{r}{p^2+r^2} \leq \frac{w_{r+1}}{w_r} < 1 + \frac{r+1}{p^2+r(r+1)}$ *(for $r = 0, 1, \ldots, p-1$)*,

(v) $\frac{w(i,j)}{\sqrt{i^2+j^2}} \geq \min\limits_{0\leq k\leq p} \frac{w_k}{\sqrt{p^2+k^2}}$ *(for all* $(i,j) \in M_p^*$*)*

all hold, then the maximum relative error of w *satisfies*

$$e \leq \max\left\{1 - \min_{0\leq k\leq p} \frac{w_k}{\sqrt{p^2+k^2}},\ \max_{0\leq r\leq p-1} H_r - 1\right\},$$

where $H_r = \sqrt{\frac{1}{p^2}\left\{(r+1)w_r - rw_{r+1}\right\}^2 + (w_{r+1}-w_r)^2}$.

The optimal neighbourhoods w_p^B, w_p^D and w_p^C are defined by the expressions (1.13)–(1.15). Using arguments similar to the ones used in Chapters 3 and 4 one can verify that these neighbourhoods satisfy inequalities (i), (ii) and (iii) of Theorem A.4. Moreover it is clear from the definitions that they satisfy inequality (v). It remains to check that inequality (iv) also holds for the optimal neighbourhoods, and then we may apply Theorem A.4 to them. The following lemma shows just that.

Lemma A.5 *Let* $p \geq 2$ *and write* $\tilde{w}_j^X = w_p^X(p,j)$ *for* $j = 0, 1, \ldots, p$, *with* $X \in \{B, C, D\}$. *Then it holds for* $r = 0, 1, \ldots, p-1$ *that*

$$1 + \frac{r}{p^2+r^2} \leq \frac{\tilde{w}_{r+1}^X}{\tilde{w}_r^X} < 1 + \frac{r+1}{p^2+r(r+1)}.$$

Proof. We first prove the statement for every possible situation, except $r = 0$ in the B-case. Apart from that exception, we always have

$$\frac{\tilde{w}_{r+1}^X}{\tilde{w}_r^X} = \frac{\sqrt{p^2+(r+1)^2}}{\sqrt{p^2+r^2}}.$$

Now the left-hand inequality follows by

$$\begin{aligned} \frac{\sqrt{p^2+(r+1)^2}}{\sqrt{p^2+r^2}} &= \frac{\sqrt{p^4+p^2(r^2+(r+1)^2)+r^2(r+1)^2}}{p^2+r^2} \\ &> \frac{\sqrt{p^4+2p^2r(r+1)+r^2(r+1)^2}}{p^2+r^2} = \frac{p^2+r(r+1)}{p^2+r^2}, \end{aligned}$$

and the right-hand inequality follows by

$$\frac{\sqrt{p^2+(r+1)^2}}{\sqrt{p^2+r^2}} = \frac{p^2+(r+1)^2}{\sqrt{p^4+p^2(r^2+(r+1)^2)+r^2(r+1)^2}} < \frac{p^2+(r+1)^2}{p^2+r(r+1)}.$$

For the remaining case ($r = 0$ and $X = B$) the statement reduces to

$$1 \leq \frac{(1-e_p^B)\sqrt{p^2+1}}{p} < \frac{p^2+1}{p^2}.$$

The right-hand inequality is true, because

$$\frac{(1-e_p^B)\sqrt{p^2+1}}{p} < \sqrt{\frac{p^2+1}{p^2}} < \frac{p^2+1}{p^2}.$$

The left-hand inequality is just $\tilde{w}_0^B \leq \tilde{w}_1^B$, which should of course be true. Substituting expression (1.10) for e_p^B, it suffices to show that

$$\left(p\sqrt{p^2+1} + 2\sqrt{p^2+1-p\sqrt{p^2+1}} - 2\right)\sqrt{p^2+1} \geq p^3. \tag{A.6}$$

For $p \geq 2$ it is always true that $\sqrt{p^2+1-p\sqrt{p^2+1}} \geq \frac{1}{2}\sqrt{2}$, so the left-hand-side of (A.6) is greater than or equal to

$$p^3 + p - (2-\sqrt{2})\sqrt{p^2+1}.$$

Now $(2-\sqrt{2})\sqrt{p^2+1} < (2-\sqrt{2})(p+1) < p$ if $p > \sqrt{2}$. In particular, it follows that (A.6) is true for $p \geq 2$. $\square$

We may thus use Theorem A.4 to get an upper bound on the maximum relative error of w_p^B, w_p^C and w_p^D. It is immediately clear that (in the notation of Lemma A.5)

$$\begin{aligned} 1 - \min_{0\leq k\leq p} \frac{\tilde{w}_k^B}{\sqrt{p^2+k^2}} &= e_p^B, \\ 1 - \min_{0\leq k\leq p} \frac{\tilde{w}_k^C}{\sqrt{p^2+k^2}} &= e_p^C, \\ 1 - \min_{0\leq k\leq p} \frac{\tilde{w}_k^D}{\sqrt{p^2+k^2}} &= 0. \end{aligned} \tag{A.7}$$

The real work lies in the evaluation of $\max_{0\leq r\leq p-1} H_r - 1$.

We first write H_r in a slightly different form:

$$H_r(w_r, w_{r+1}) = \sqrt{\frac{p^2+(r+1)^2}{p^2}w_r^2 - \frac{p^2+r(r+1)}{p^2}2w_r w_{r+1} + \frac{p^2+r^2}{p^2}w_{r+1}^2}. \tag{A.8}$$

It is a straightforward – albeit slightly tedious – task to work out the value of H_r for $r = 0, \ldots, p-1$ in all three cases. Just as in the proof of Lemma A.5, we can use the same approach in almost every situation, the only exception being (again) $r = 0$ in the B-case.

To save some space we define a function $q_p(x)$ as follows:

$$\begin{aligned} q_p(x) := \; & \left(p^2+x^2\right)\left(p^2+(x+1)^2\right) \\ & - \left(p^2+x(x+1)\right)\sqrt{(p^2+x^2)\,(p^2+(x+1)^2)}. \end{aligned}$$

Then we obtain the following expressions:

$$\begin{aligned} H_r(\tilde{w}_r^B, \tilde{w}_{r+1}^B) &= \sqrt{\tfrac{2(1-e_p^B)^2 q_p(r)}{p^2}} \qquad (r = 1, \dots, p-1), \\ H_r(\tilde{w}_r^C, \tilde{w}_{r+1}^C) &= \sqrt{\tfrac{2(1-e_p^C)^2 q_p(r)}{p^2}} \qquad (r = 0, \dots, p-1), \\ H_r(\tilde{w}_r^D, \tilde{w}_{r+1}^D) &= \sqrt{\tfrac{2 q_p(r)}{p^2}} \qquad (r = 0, \dots, p-1). \end{aligned}$$

For every $p \neq 0$ the function $q_p(x)$ is monotone decreasing on $\left[-\frac{1}{2}, \infty\right)$. Therefore:

$$\begin{aligned} \max_{0 \le r \le p-1} H_r(\tilde{w}_r^B, \tilde{w}_{r+1}^B) &= \max\left\{H_0(\tilde{w}_0^B, \tilde{w}_1^B),\ H_1(\tilde{w}_1^B, \tilde{w}_2^B)\right\}, \\ \max_{0 \le r \le p-1} H_r(\tilde{w}_r^C, \tilde{w}_{r+1}^C) &= H_0(\tilde{w}_0^C, \tilde{w}_1^C), \\ \max_{0 \le r \le p-1} H_r(\tilde{w}_r^D, \tilde{w}_{r+1}^D) &= H_0(\tilde{w}_0^D, \tilde{w}_1^D). \end{aligned} \tag{A.9}$$

Using the fact that $q_p(0) = p^2\left(p^2 + 1 - p\sqrt{p^2+1}\right)$, we have

$$\begin{aligned} H_0(\tilde{w}_0^C, \tilde{w}_1^C) &= (1 - e_p^C)\sqrt{2\left(p^2 + 1 - p\sqrt{p^2+1}\right)} \\ &= \frac{2\sqrt{2p^2 + 2 - 2p\sqrt{p^2+1}}}{\sqrt{2p^2 + 2 - 2p\sqrt{p^2+1}} + 1} = 1 + e_p^C \end{aligned}$$

and

$$H_0(\tilde{w}_0^D, \tilde{w}_1^D) = \sqrt{2\left(p^2 + 1 - p\sqrt{p^2+1}\right)} = 1 + e_p^D.$$

It follows by (A.9) and Theorem A.4 that w_p^C and w_p^D have maximum relative errors less than or equal to e_p^C and e_p^D, respectively. But we know from [11] that these are the actual values of the maximum relative error. Therefore the upper bound on the maximum relative error provided by Theorem A.4 is exact for the optimal neighbourhoods w_p^C and w_p^D.

It remains to proof the same result for w_p^B. We need to show two things: $H_0(\tilde{w}_0^B, \tilde{w}_1^B) = 1 + e_p^B$ and $H_0(\tilde{w}_0^B, \tilde{w}_1^B) \ge H_1(\tilde{w}_1^B, \tilde{w}_2^B)$. For the first part we have no option but to verify by brute force that (using (1.13) and (A.8))

$$p^2 + 1 - 2(1 - e_p^B)p\sqrt{p^2+1} + (1 - e_p^B)^2(p^2+1) = (1 + e_p^B)^2.$$

Having established this, the second part can be proved as follows. We know that

$$\left(H_0(\tilde{w}_0^B, \tilde{w}_1^B)\right)^2 = \left\{1 + (1 - e_p^B)^2\right\}(p^2+1) - 2(1 - e_p^B)p\sqrt{p^2+1}.$$

Because $1 + (1 - e_p^B)^2 > 2(1 - e_p^B)$, this means that

$$\begin{aligned} \left(H_0(\tilde{w}_0^B, \tilde{w}_1^B)\right)^2 &> 2(1 - e_p^B)\left(p^2 + 1 - p\sqrt{p^2+1}\right) \\ &> 2(1 - e_p^B)^2\left(p^2 + 1 - p\sqrt{p^2+1}\right) \end{aligned}$$

Therefore $H_0(\tilde{w}_0^B, \tilde{w}_1^B) > \sqrt{\frac{2(1-e_p^B)^2 q_p(0)}{p^2}} > \sqrt{\frac{2(1-e_p^B)^2 q_p(1)}{p^2}} = H_1(\tilde{w}_1^B, \tilde{w}_2^B)$. And it follows that the upper bound on the maximum relative error provided by Theorem A.4 is also exact for the optimal neighbourhood w_p^B.

A.4 Proof of Lemma 3.7

Lemma 3.7 *Let $p \geq 2$. For every $n \geq 1$ the neighbourhoods ${}_n\mathcal{N}_p^B$, ${}_n\mathcal{N}_p^C$ and ${}_n\mathcal{N}_p^D$ defined in Proposition* 3.6 *satisfy $(r+1)n_r > rn_{r+1}$ (for $r = 0, \ldots, p-1$).*

Proof. For $r = 0$ the condition trivially holds in all cases, so we take $r \geq 1$. For ${}_n\mathcal{N}_p^B$ we have $n_0 = np$, $n_j = \langle n\left(1 - e_p^B\right)\sqrt{p^2 + j^2}\rangle$ $(j = 1, \ldots, p)$. We must check that

$$(r+1)\langle n\left(1 - e_p^B\right)\sqrt{p^2 + r^2}\rangle - r\langle n\left(1 - e_p^B\right)\sqrt{p^2 + (r+1)^2}\rangle > 0.$$

Since $x - \frac{1}{2} < \langle x\rangle \leq x + \frac{1}{2}$ it suffices that

$$n\left(1 - e_p^B\right)\left\{(r+1)\sqrt{p^2 + r^2} - r\sqrt{p^2 + (r+1)^2}\right\} > 1.$$

For every $p \geq 2$ the function

$$f_1(x) := (x+1)\sqrt{p^2 + x^2} - x\sqrt{p^2 + (x+1)^2}$$

is monotone decreasing to 0 on $\left[-\frac{1}{2}, \infty\right)$. Thus it is sufficient to prove that the condition holds for $r = p - 1$, i.e.

$$n\left(1 - e_p^B\right)\left\{p\sqrt{p^2 + (p-1)^2} - p(p-1)\sqrt{2}\right\} > 1.$$

Inserting expression (1.10) for e_p^B, we must check that

$$\frac{1}{p}\left\{2\sqrt{p^2 + 1 - p\sqrt{p^2+1}} + p\sqrt{p^2+1} - 2\right\}\left\{\sqrt{p^2 + (p-1)^2} - (p-1)\sqrt{2}\right\} > \frac{1}{n}$$

The function

$$f_2(x) := \frac{1}{x}\left\{2\sqrt{x^2 + 1 - x\sqrt{x^2+1}} + x\sqrt{x^2+1} - 2\right\}\left\{\sqrt{x^2 + (x-1)^2} - (x-1)\sqrt{2}\right\}$$

is monotone increasing for $x \geq 2$. Since $f_2(2) \approx 1.61$, we find that $f_2(p) \geq f_2(2) > \frac{1}{n}$ for every $p \geq 2$ and every $n \geq 1$. Thus (2.10) indeed holds.

For ${}_n\mathcal{N}_p^C$ we have $n_j = \langle n\left(1 - e_p^C\right)\sqrt{p^2 + j^2}\rangle$ $(j = 1, \ldots, p)$. Using the same argument as above, it suffices to check that

$$n\left(1 - e_p^C\right)\left\{p\sqrt{p^2 + (p-1)^2} - p(p-1)\sqrt{2}\right\} > 1.$$

Inserting expression (1.12) for e_p^C, this becomes

$$\frac{2\left\{p\sqrt{p^2+(p-1)^2}-p(p-1)\sqrt{2}\right\}}{\sqrt{2p^2+2-2p\sqrt{p^2+1}}+1} > \frac{1}{n}.$$

The function

$$f_3(x) := \frac{x\sqrt{x^2+(x-1)^2}-x(x-1)\sqrt{2}}{\sqrt{2x^2+2-2x\sqrt{x^2+1}}+1}$$

is monotone increasing for $x \geq 2$. Since $f_3(2) \approx 0.81 > \frac{1}{2n}$, the result follows for every $p \geq 2$ and $n \geq 1$.

Finally, for ${}_n\mathcal{N}_p^D$ we have $n_j = \lceil n\sqrt{p^2+j^2}\rceil$ $(j = 1, \ldots, p)$ and we must verify that

$$(r+1)\lceil n\sqrt{p^2+r^2}\rceil - r\lceil n\sqrt{p^2+(r+1)^2}\rceil > 0.$$

By $x \leq \lceil x\rceil < x+1$, it suffices to check that

$$n\left\{(r+1)\sqrt{p^2+r^2} - r\sqrt{p^2+(r+1)^2}\right\} > 1.$$

Again using the fact that f_1 is monotone decreasing, we can restrict our attention to the case $r = p-1$, i.e. check that

$$\left\{p\sqrt{p^2+(p-1)^2}-p(p-1)\sqrt{2}\right\} > \frac{1}{n}.$$

The function

$$f_4(x) := x\sqrt{x^2+(x-1)^2}-x(x-1)\sqrt{2}$$

is monotone increasing for $x \geq 2$. Since $f_4(2) \approx 1.64 > \frac{1}{n}$, we are done for every $p \geq 2$ and $n \geq 1$. □

A.5 Proof of Lemma 4.6

Lemma 4.6 *Let $p \geq 2$. For every $n \geq 1$ the neighbourhoods ${}_n^*\mathcal{N}_p^B$ and ${}_n^*\mathcal{N}_p^C$ defined in Proposition 4.5 satisfy $(r+1)n_r > rn_{r+1}$ (for $r = 0, \ldots, p-1$).*

Proof. For $r = 0$ the condition trivially holds in both cases. For ${}_n^*\mathcal{N}_p^B$ we have $n_0 = np$, $n_1 = \langle n\left(1-e_p^B\right)\sqrt{p^2+1}\rangle$ and $n_j = \lceil n\left(1-e_p^B\right)\sqrt{p^2+j^2}\rceil$ $(j = 2, \ldots, p)$. First take $r = 1$. We have to show that

$$2\langle n\left(1-e_p^B\right)\sqrt{p^2+1}\rangle - \lceil n\left(1-e_p^B\right)\sqrt{p^2+4}\rceil > 0.$$

Since $x - \frac{1}{2} < \langle x\rangle \leq x + \frac{1}{2}$ and $x \leq \lceil x\rceil < x+1$, it is sufficient to show that

$$n\left(1-e_p^B\right)\left(2\sqrt{p^2+1}-\sqrt{p^2+4}\right) > 2.$$

Substituting expression (1.10) for e_p^B, we must check that

$$\frac{2\sqrt{p^2+1-p\sqrt{p^2+1}}+p\sqrt{p^2+1}-2}{p^2}\left(2\sqrt{p^2+1}-\sqrt{p^2+4}\right)>\frac{2}{n}$$

The function

$$f_5(x) := \frac{2\sqrt{x^2+1-x\sqrt{x^2+1}}+x\sqrt{x^2+1}-2}{x^2}\left(2\sqrt{x^2+1}-\sqrt{x^2+4}\right)$$

is monotone increasing for $x \geq 2$. Note that $f_5(2) \approx 1.61$ and $f_5(3) \approx 2.69$. For $p \geq 3$ the statement now follows since $f_5(p) \geq f_5(3) > \frac{2}{n}$ for every $n \geq 1$. For $p = 2$ we have $f_5(2) > \frac{2}{n}$ for every $n \geq 2$, and the reader may verify that the statement also holds for the special case $p = 2$, $n = 1$.

Now take $r \geq 2$. We have to check that

$$(r+1)\lceil n\left(1-e_p^B\right)\sqrt{p^2+r^2}\rceil - r\lceil n\left(1-e_p^B\right)\sqrt{p^2+(r+1)^2}\rceil > 0.$$

Following the argument used in the D-case in the proof of Lemma 3.7, it is sufficient to show that

$$\left(1-e_p^B\right)\left\{p\sqrt{p^2+(p-1)^2}-p(p-1)\sqrt{2}\right\} > \frac{1}{n}.$$

We already verified this in the proof of Lemma 3.7.

For ${}_n^*\mathcal{N}_p^C$ we have $n_j = \langle n\left(1-e_p^C\right)\sqrt{p^2+j^2}\rangle$ $(j = 0, 1)$ and $n_j = \lceil n\left(1-e_p^C\right)\sqrt{p^2+j^2}\rceil$ $(j = 2, \ldots, p)$. First take $r = 1$. We have to check that

$$2\langle n\left(1-e_p^C\right)\sqrt{p^2+1}\rangle - \lceil n\left(1-e_p^C\right)\sqrt{p^2+4}\rceil > 0.$$

Since $x - \frac{1}{2} < \langle x\rangle \leq x + \frac{1}{2}$ and $x \leq \lceil x\rceil < x + 1$, it is sufficient to show that

$$n\left(1-e_p^C\right)\left(2\sqrt{p^2+1}-\sqrt{p^2+4}\right) > 2.$$

Substituting expression (1.12) for e_p^C, we must check that

$$\frac{2\sqrt{p^2+1}-\sqrt{p^2+4}}{\sqrt{2p^2+2-2p\sqrt{p^2+1}}+1} > \frac{1}{n}.$$

The function

$$f_6(x) := \frac{2\sqrt{x^2+1}-\sqrt{x^2+4}}{\sqrt{2x^2+2-2x\sqrt{x^2+1}}+1}$$

is monotone increasing for $x \geq 2$. In particular $f_6(2) \approx 0.81$ and $f_6(3) \approx 1.35$. The statement now follows for $p \geq 3$ since $f_6(p) \geq f_6(3) > \frac{1}{n}$ for every $n \geq 1$. For $p = 2$ we have $f_6(2) > \frac{1}{n}$ for every $n \geq 2$, and the reader may easily verify that the statement also holds for the special case $p = 2$, $n = 1$.

For $r \geq 2$ we have to check that

$$(r+1)\lceil n\left(1-e_p^C\right)\sqrt{p^2+r^2}\rceil - r\lceil n\left(1-e_p^C\right)\sqrt{p^2+(r+1)^2}\rceil > 0.$$

As before it is sufficient to show that

$$\left(1-e_p^C\right)\left\{p\sqrt{p^2+(p-1)^2} - p(p-1)\sqrt{2}\right\} > \frac{1}{n},$$

and we already verified this in the proof of Lemma 3.7. □

Appendix B

Programs

This appendix contains pseudocodes that may be used to calculate the results of Chapter 5. We restrict ourselves to the case $p \geq 2$, since the implementation of the theory for $p = 1$ is straightforward.

B.1 A program that constructs ${}_nN_p^B$

```
for i = 1, 2, ..., p
    N(±i, 0) := n * i =: N(0, ±i);
end
μ := n;
for j = 1, 2, ..., p
    N(±p, ±j) := round(n * (1 − e_p^B) * sqrt(p^2 + j^2));
    a := N(p, j)/sqrt(p^2 + j^2);
    if a < μ
        μ := a;
    end
end
for all other (i, j) ∈ M_p^*
    N(i, j) := ceil(μ * sqrt(i^2 + j^2));
end
```

B.2 A program that constructs ${}_n^*N_p^B$

```
for i = 1, 2, ..., p
    N(±i, 0) := n * i =: N(0, ±i);
end
μ := n;
for j = 1, ..., p
    if j = 1
        N(±p, ±j) := round(n * (1 − e_p^B) * sqrt(p^2 + j^2));
```

```
    else
        N(±p, ±j) := ceil(n * (1 − e_p^B) * sqrt(p^2 + j^2));
    end
    a := N(p, j)/sqrt(p^2 + j^2);
    if a < μ
        μ := a;
    end
end
for all other (i, j) ∈ M_p^*
    N(i, j) := ceil(μ * sqrt(i^2 + j^2));
end
```

B.3 A program that constructs ${}_nN_p^C$

```
μ := n;
for j = 0, 1, . . . , p
    N(±p, ±j) := round(n * (1 − e_p^C) * sqrt(p^2 + j^2));
    a := N(p, j)/sqrt(p^2 + j^2);
    if a < μ
        μ := a;
    end
end
for all other (i, j) ∈ M_p^*
    N(i, j) := ceil(μ * sqrt(i^2 + j^2));
end
```

B.4 A program that constructs ${}_n^*N_p^C$

```
μ := n;
for j = 0, 1, . . . , p
    if j = 0, 1
        N(±p, ±j) := round(n * (1 − e_p^C) * sqrt(p^2 + j^2));
    else
        N(±p, ±j) := ceil(n * (1 − e_p^C) * sqrt(p^2 + j^2));
    end
    a := N(p, j)/sqrt(p^2 + j^2);
    if a < μ
        μ := a;
    end
end
for all other (i, j) ∈ M_p^*
    N(i, j) := ceil(μ * sqrt(i^2 + j^2));
end
```

B.5 A program that constructs ${}_nN_p^D$

for $i = 1, 2, \ldots, p$
  $N(\pm i, 0) := n * i =: N(0, \pm i);$
end
for all other $(i, j) \in M_p^*$
  $N(i, j) :=$ **ceil**$(n *$ **sqrt**$(i^2 + j^2));$
end

B.6 A program that computes the maximum relative error

for $r = 0, 1, \ldots, p-1$
  $t :=$ **floor**$((p^2 * (N(p, r+1) - N(p, r))/((r+1) * N(p, r) - r * N(p, r+1)));$
  if $t < r$
    $H(r) := N(p, r)/$**sqrt**$(p^2 + r^2);$
  elseif $t = r$
    $H(r) :=$ **sqrt**$((1/p^2) * ((r+1) * N(p, r) - r * N(p, r+1))^2$
      $+(N(p, r+1) - N(p, r))^2);$
  elseif $t > r$
    $H(r) := N(p, r+1)/$**sqrt**$(p^2 + (r+1)^2);$
  end
end
$c(1) :=$ **max**$(H(0), \ldots, H(p-1));$
$c(2) := N(p, 0)/p;$
for $k = 1, 2, \ldots, p$
  $a := N(p, k)/$**sqrt**$(p^2 + k^2);$
  if $a < c(2)$
    $c(2) := a;$
  end
end
$s := n;$
$e :=$ **max**$(1 - c(2)/s, c(1)/s - 1);$

In the C-case, the last two lines of this program are replaced by:
$s := (1/2) * (c(1) + c(2));$
$e := (c(1) - c(2))/(c(1) + c(2));$

In the D-case, the for-loop that computes $c(2)$ may be dropped since it yields no change.

Bibliography

[1] A.L.D. Beckers, A.W.M. Smeulders – A comment on "A note on 'Distance transformations in digital images' ". *Computer Vision, Graphics, and Image Processing* **47** (1989), 89-91. Comments on [23].

[2] G. Borgefors – Distance transformations in arbitrary dimensions. *Computer Vision, Graphics, and Image Processing* **27** (1984), 321-345.

[3] G. Borgefors – Distance transformations in digital images. *Computer Vision, Graphics, and Image Processing* **34** (1986), 344-371.

[4] G. Borgefors – Hierarchical chamfer matching: a parametric edge matching algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **10** (1988), 849-865.

[5] G. Borgefors – Another comment on "A note on 'Distance transformations in digital images' ". *CVGIP: Image Understanding* **54** (1991), 301-306. Comments on [23].

[6] M.A. Butt, P. Maragos – Optimum design of chamfer distance transforms. *IEEE Transactions on Image Processing* **7** (1998), 1477-1484.

[7] Y. Chehadeh – *Opérateurs locaux de distance en maillages rectangulaire et parallélépipédique: application à l'analyse d'images.* PhD Thesis, 1997, Université de Savoie. Downloadable from:
`www.univ-savoie.fr/Portail/Groupes/LISTIC/Theses/thesechehadeh.pdf`

[8] D. Coquin, Ph. Bolon – Discrete distance operator on rectangular grids. *Pattern Recognition Letters* **16** (1995), 911-923.

[9] O. Cuisenaire – *Distance Transformations: Fast Algorithms and Applications to Medical Image Processing.* PhD Thesis, 1999, Université Catholique de Louvain. Downloadable from:
`ltswww.epfl.ch/~cuisenai/papers/oc_thesis.pdf`

[10] P.E. Danielsson – Euclidean distance mapping. *Computer Graphics and Image Processing* **14** (1980), 227-248.

[11] A. Hajdu, L. Hajdu, R. Tijdeman – Approximation of Euclidean distances by chamfer distances. Unpublished.

[12] F. Leymarie, M.D. Levine – Fast raster scan distance propagation on the discrete rectangular lattice. *CVGIP: Image Understanding* **55** (1992), 84-94.

[13] G. Malandain, C. Fouard – *On optimal chamfer masks and coefficients*. Rapport de recherche, 2005, Institut National de Recherche en Informatique et en Automatique. Downloadable from: `www.inria.fr/rrrt/rr-5566.html`

[14] U. Montanari – A method for obtaining skeletons using a quasi-Euclidean distance. *Journal of the Association for Computing Machinery* **15** (1968), 600-624.

[15] A. Rosenfeld, J.L. Pfaltz – Sequential operations in digital picture processing. *Journal of the Association for Computing Machinery* **13** (1966), 471-494.

[16] A. Rosenfeld, J.L. Pfaltz – Distance functions on digital pictures. *Pattern Recognition* **1** (1968), 33-61.

[17] S. Scholtus, R. Tijdeman – Chamfer distances with integer neighborhoods. Downloadable from: `www.math.leidenuniv.nl/~scholtus/chamfer.pdf`

[18] E. Thiel – *Les distances de chanfrein en analyse d'images: fondements et applications*. PhD Thesis, 1994, Université Joseph Fourier de Grenoble. Downloadable from: `www.lis.inpg.fr/stages_dea_theses/theses/manuscript/thiel.pdf`

[19] E. Thiel, A. Montanvert – Etude et amélioration des distances du chanfrein pour l'analyse d'images. *Technique et Science Informatiques* **11** (1992), 9-41. Downloadable from: `www.lim.univ-mrs.fr/~thiel/print/1992-tsi-thiel-montanvert.pdf`

[20] E. Thiel, A. Montanvert – Chamfer masks: discrete distance functions, geometrical properties and optimization. *Proceedings of the 11th International Conference on Pattern Recognition* **3** (1992), 244-247. Summarises part of the results of [19] in English. Downloadable from: `www.lim.univ-mrs.fr/~thiel/print/1992-icpr11-thiel-montanvert.pdf`

[21] B.H.J. Verwer – *Distance Transforms: Metrics, Algorithms and Applications*. PhD Thesis, 1991, Technische Universiteit Delft.

[22] B.H.J. Verwer – Local distances for distance transformations in two and three dimensions. *Pattern Recognition Letters* **12** (1991), 671-682. Corresponds with Chapter 3 of [21].

[23] A.M. Vossepoel – A note on “Distance transformations in digital images”. *Computer Vision, Graphics, and Image Processing* **43** (1988), 88-97. Comments on [3].

[24] M. Yamashita, T. Ibaraki – Distances defined by neighbourhood sequences. *Pattern Recognition* **19** (1986), 237-246.

Acknowledgements

I would like to thank Professor Tijdeman for his excellent supervision of the thesis. His remarks were of great help to me, both during the research and the editing of the manuscript.

I would also like to thank Professor Kallenberg and Doctor Evertse for accepting to proof-read the thesis and acting as members of the exam committee.

Finally I would like to thank my parents and my brother for supporting me and for humouring me when I overenthousiastically tried to explain the solution of some mathematical problem to them.