Detection, extraction, and analysis of the vein network of the slime mould *Physarum polycephalum*

Werner Baumgarten and Marcus J.B. Hauser

ABSTRACT

The plasmodium of the slime mould *Physarum polycephalum* forms an extended and complex two-dimensional vein network that is used to transport protoplasm through the giant cell. To obtain insights into the topology of this vein network, its graph structure needs to be extracted from experimental images. This amounts to the task of extracting and reconstructing very tiny venules (elements) from an image that covers a relatively huge area. A protocol containing a sequence of image processing and correction steps that allows a highly accurate detection and extraction of the underlying graph has been developed and is described in detail. The analysis of the extracted data reveals that the veins and venules of the plasmodium of *P. polycephalum* form a regular, cubic graph.

Keywords: real world network, regular graph, *Physarum polycephalum*, data extraction.

Abteilung Biophysik, Institut für Experimentelle Physik, Otto-von-Guericke-Universität Magdeburg, 39106 Magdeburg, Germany.
E-mails: werner.baumgarten@ovgu.de, marcus.hauser@ovgu.de

1   INTRODUCTION

Transportation networks are of enormous importance in biological and technical systems as well as in everyday life, and have, therefore, attracted substantial scientific research interest. During the last decades, the topological structure of such transportation networks has been in the focus of numerous studies. It has been found that many of these networks possess features that are typical for 'complex dynamic networks', such as small-world properties or scale-free behaviour [1, 2, 3]. This means that the individual nodes of the networks have different connectivities, and that such networks may possess some highly connected nodes that act as hubs, as well known from air-transportation networks [4, 5].

Generally, in studies on complex real world networks the networks are obtained from data already available in electronic form in large databases. This approach has successfully been used to obtain and analyse real world networks as diverse as transportation (like the air-transportation [4, 5], railway [6], urban railway networks [7], and electric power grids [1]), economic [8], and social networks (like scientific co-author [9] or co-staring film actors networks [1]). For an overview, see for instance references [2, 3].

While networks, whose data are readily available in electronic form are frequently studied, the situation changes dramatically when it comes to networks, whose graph structure data must be first measured, digitised, and extracted from images or photographs, before it may finally be analysed. Examples of such data are blood vascularisation networks [10] and insect trails and galleries [11, 12], as well as many types of data collected by monitoring satellites.

The challenge of extracting the graph underlying such networks stems from the fact that one needs to follow the development of the network over time in an area that is relatively large compared to the size of the entities (veins, trails, tunnels, etc.) that form the network. This means that well developed imaging techniques and analysis tools [13] may be required. Often, however, a combination of methods is necessary which is optimised for the detection and high-accuracy reconstruction of data on multiple scales.

Such a situation is encountered when it comes to the analysis of the vein network formed by the plasmodium of the acellular slime mould *Physarum polycephalum*. This slime mould (myxomycete) forms a giant single cell, a so-called syncytium, whose extension may even reach some square meters. This cell forms a vein network, through which its protoplasm is transpor-

ted to and fro in a regular basis [14, 15]. It has been shown that the plasmodium is able to optimise the structure of its vein network [15, 16]. In the presence of multiple food sources, the cell tries to connect these food sources through its vein network, however at a minimum cost. This is achieved by shrinking all veins other than those providing the direct connection between the food sources. This property of *P. polycephalum* has been exploited to let the amoeboid slime mould solve a maze [17] or other graph theoretical problems, like constructing Steiner minimum pathway trees [18], or finding minimum-risk pathways in a situation, where the slime mould is subjected to two competing risk (or cost) factors [19].

Surprisingly, in spite of its ability to solve graph theoretical tasks, the graph properties of the plasmodial vein network of *P. polycephalum* have not been addressed so far. In earlier papers, the graph is described colloquially as a more or less regular vein network [20], while recently, it has been modelled as a weighted planar network whose central nodes have a somewhat lower connectivity than more peripheral nodes [21].

In the present article, we report on methods to detect and extract the vein network from *P. polycephalum* from acquired experimental images. The objective is to obtain the graph structure of this vein network at large scales, while individual venules might be very tiny. The paper is organised as follows: The next section contains the experimental methods, while section 3 reports on the individual steps of image and data processing applied to the data. Finally, in section 4 we subject the extracted skeletonised network to graph analysis, thus providing a proof of concept that the graph characteristics may be extracted and determined from our data. Last but not least, we finish the article with a brief Discussion.

2   EXPERIMENTAL METHODS

Sclerotia of the *Physarum polycephalum* strain HU195 × HU200 [22] were obtained from plasmodia grown on moist filter paper and fed with oat flakes (Kölln Flocken) at 21°C in the dark. The cellular material on filter paper was dried and stored in the dark, thus inducing the transition to sclerotia. The sclerotium is a dormant, dehydrated form of *P. polycephalum* that may convert back into the plasmodium whence the environmental conditions turn favourable again.

Experiments were started by placing a rectangular filter paper stripe containing a sclerotium on a 1% (w/v) agar gel close to the edge of a covered 10 × 10 cm² square Petri dish. About 25000–26000 s later, the *P. polycephalum* develops back into a

plasmodium which starts to explore the gel in the Petri dish. This growing plasmodium expands over the gel layer and begins to form a network of veins and venules.

The Petri dish was illuminated homogeneously from below by a cold light source emitting white light and monitored from above by a 8 bit CCD camera (Hamamatsu C3077). The images were collected with a resolution of $768 \times 576 \, \text{pix}^2$ corresponding to an area of $61.5 \times 46.1 \, \text{mm}^2$ (i.e. a resolution of $0.08 \, \text{mm pix}^{-1}$) at a sampling frequency of 0.1 Hz. The frames contain grey-value images that have 256 grey values. The images were transferred to a computer for later data processing. This comprises the detection and extraction of the network from the acquired images. The processed data are used for the analysis of the properties of the plasmodial vein network.

3   IMAGE PROCESSING

To extract the topological structure of the plasmodial vein network of *Physarum polycephalum* from the acquired images, a series of image processing operations and corrections must be performed. These manipulations are compiled in the flow chart presented in Figure 1. Concomitantly, the network, as it appears after the most prominent of these manipulations, is depicted in Figure 2.

3.1   Background subtraction

To remove constant structures and possible illumination gradients from the images, all acquired images are background-subtracted. The image used as 'background image' is the last frame acquired before the plasmodium begins to invade the area of observation. This image is subtracted from all further images, yielding the background-subtracted data set (Fig. 2).

3.2   Binarisation

In the next step, the background-subtracted images are binarised. Due to the occurrence of some very thin venules in the vein network, any possible choice of threshold value may lead to the 'loss' of a few pixels in the binarised images. In other words, the grey values of some pixels of venules lay just below the binarisation threshold. Therefore, a series of correction algorithms have to be applied to repair these mistakes.

3.3   Correction algorithms for binarised pictures

Six correction operations were applied to the binarised data in order to account for and to reinsert some missing pixels, which have just fallen short at the binarisation process. Hence, the corrections serve two purposes. At the one hand, they attempt to reconnect venules that were 'erroneously' ruptured by the image processing steps. At the other hand, some of the corrections are necessary to satisfy the algorithmic requirements for computing the length of vein segments (section 3.5.2).

The vicinity of every central black pixel (which belongs to the graph) is analysed using a sliding 3×3 or $5 \times 5 \, \text{pix}^2$ window. In this window, the neighbourhood of the central pixel is examined, and depending on the nature of its surrounding, one of the following six correction algorithms may be applied.
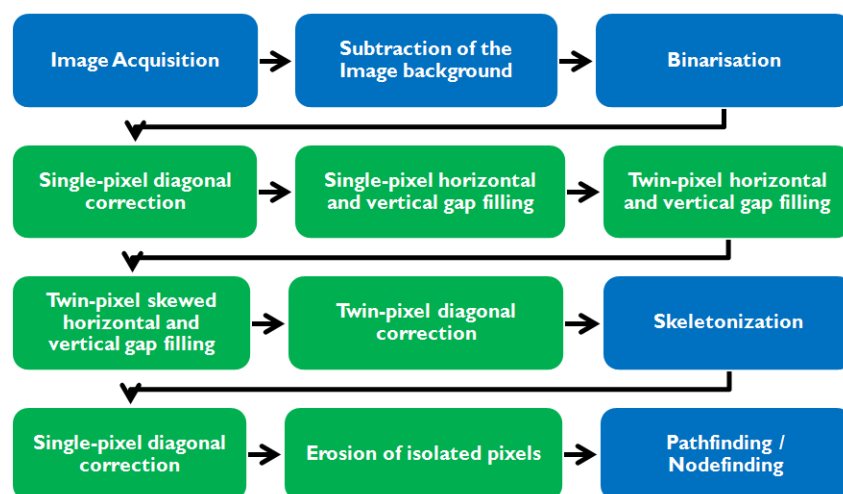

Figure 1 – Sequence of data processing steps applied to the acquired images of the growing vein network of *Physarum polycephalum*. The blue boxes represent image processing operations while the green boxes represent corrections to the binarised pictures.
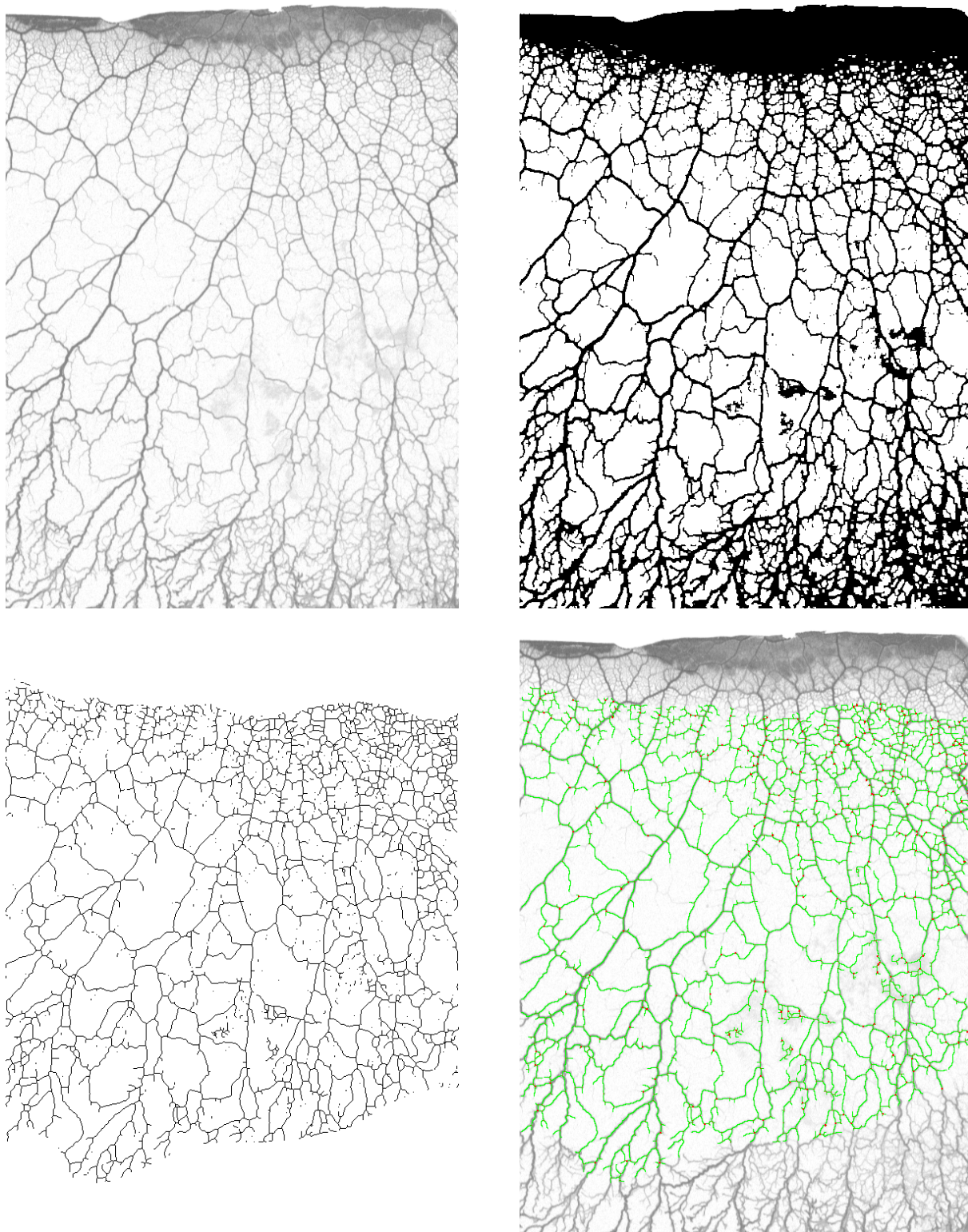
Figure 2 – Plasmodial vein network of *Physarum polycephalum* 62000 s after the start of the experiment. **(top left)** Background-subtracted image; **(top right)** Binarised network after all corrections, as presented in Figure 1; **(bottom left)** Skeleton of the binarised network; **(bottom right)** Superimposition of the extracted skeletonised graph on the background-subtracted network. The recognised segments and nodes of the graph are labelled in green and red, respectively, and superimposed on the vein network. Dimensions: 61.5×46.1 mm^2.

3.3.1 Single-pixel diagonal correction (1DC)

The single-pixel diagonal correction (1DC) is solely required due to requirements originating from the length measurement of the veins and venules (section 3.5.2). Since this algorithm operates only along the horizontal and vertical directions of the pixel matrix, diagonal venules must be corrected by turning one pixel in the adjacent off-diagonal black, as demonstrated in Figure 3a.
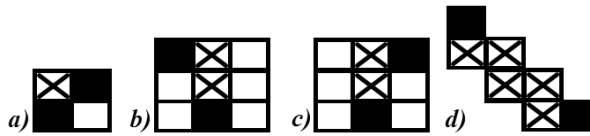

Figure 3 – Schematic drawing of the corrections performed by **(a)** the single-pixel diagonal correction (1DC), **(b),(c)** the twin-pixel skewed horizontal and vertical gap filling (2SC), and **(d)** the twin-pixel diagonal correction (2DC) algorithms. In all pictures the crosses mark the operations where white pixels will become black. Cases that are symmetric to the ones shown here exist, but they are not presented in the Figure.

3.3.2 Single-pixel horizontal and vertical gap filling (1HC)

Often, single faint pixels belonging to a venule are not recognised in the graph. To correct for such one-pixel gaps, gap filling algorithms were used, which scan the matrix of pixels in horizontal and vertical directions. Whenever a white pixel is encountered in between two black pixels, the white pixel is converted into a black one, thus reconnecting the segments of the venule.

3.3.3 Twin-pixel horizontal and vertical gap filling (2HC)

In analogy to the 1HC algorithm, the 2HC algorithm reconnects two horizontally or vertically oriented black pixels by transforming the two intermittent pixels into black ones.

Note that these operations are not effectuated if the missing sequence exceeds two pixels, i.e., if there are more than two white pixels in between two black pixels. This limit is implemented to avoid the formation of spurious edges in the graph.

3.3.4 Twin-pixel skewed horizontal and vertical gap filling (2SC)

The gap filling provided by the 1HC algorithm is restricted to pixels, where a black, a white, and another black pixel are perfectly aligned horizontally or vertically. However, the veins of the network grow at any possible angle with respect to the orientation of the array of pixels. To account for a sequence of pixels that

are 'tilted' or 'skewed', so that they lay slightly off the horizontal or vertical, the 2SC algorithm was developed. It deals with situations as depicted in Figure 3b,c, and it provides a correction by connecting the black pixels as shown in Figure 3b,c. Note that there are additional permutations of that theme which are accessible by rotation of the positions of the black pixels.

3.3.5 Twin-pixel diagonal correction (2DC)

As the 2HC algorithm is the two-pixel extension of 1HC, the twin-pixel diagonal correction (2DC) is the extension of 2SC. Here, diagonal gaps of two white pixels lying between two black pixels are corrected. Due to the necessary presence of horizontal and vertical connections between the black pixels in order to count the pixels and hence to determine the length of a segment (section 3.5.2), a total of 5 pixels are turned into black ones, as shown in Figure 3d.

3.3.6 Erosion of isolated pixels (PX)

Last but not least, isolated pixels are deleted. If a black central point has no neighbours belonging to a vein or venule, it is considered spurious and deleted (i.e. it becomes white).

3.3.7 Effect of the sequence of correction algorithms

For all corrections performed, any pixel that was added by a correcting operation is marked and excluded from participating at further corrections. Nevertheless, it is important to note that the sequence by which these correction algorithms are applied is not commutative. To determine the best sequence of the correction steps, their sequential order was permutated. As seen from Table 1, which compiles the individual permutations studied, the first and last correction steps are the same for all permutations: the first step always is the single-pixel diagonal correction (1DC), while the last always is the erosion of isolated pixels (PX).

Two criteria are used to determine the quality of the individual sequences of correction algorithms. The first criterium is a visual comparison, whether the reconstructed and corrected network still matches to the photographed network as depicted in the original or background-subtracted images. The second criterium demands that the number of correction steps effectuated at the skeletonised graph be as low as possible (Fig. 4). In other words, a low abundance of corrections is preferred, in order to avoid any equivocal corrections.

The abundance of the performed corrections steps was determined for each of the six permutations of Table 1 and is shown
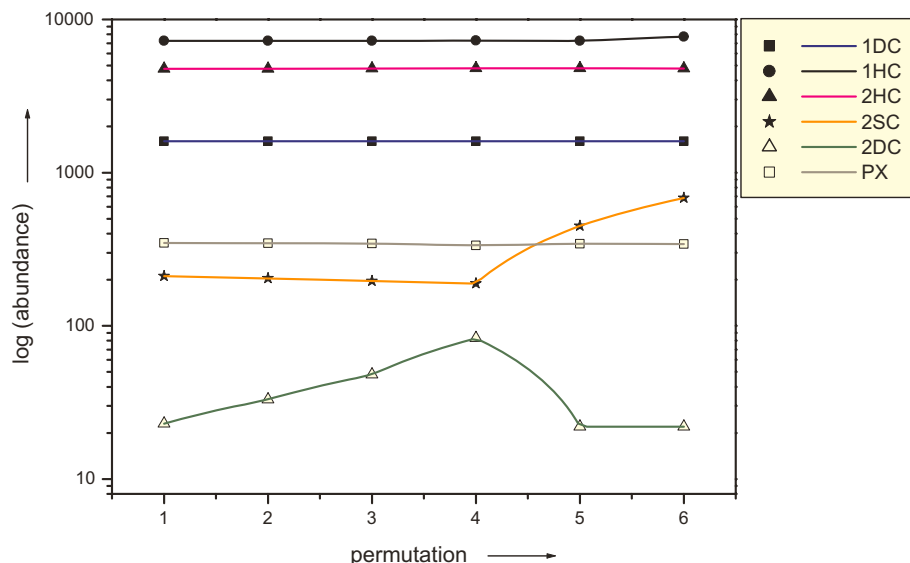
Figure 4 – Abundance of correction steps for the different permutations of the correction algorithms, as compiled in Table 1. Note that the number of corrections is plotted on a logarithmic scale.

Table 1 – Permutations of the application sequence of the correction algorithms. The acronyms stand for: 1DC single-pixel diagonal correction; 1HC single-pixel horizontal and vertical gap filling; 2HC twin-pixel horizontal and vertical gap filling; 2SC twin-pixel skewed horizontal and vertical gap filling; 2DC twin-pixel diagonal correction; and PX erosion of isolated pixels.

Permutation 1	Permutation 2	Permutation 3	Permutation 4	Permutation 5	Permutation 6
1DC	1DC	1DC	1DC	1DC	1DC
1HC	1HC	1HC	2DC	1HD	2SC
2HC	2HC	2DC	1HC	2SC	1HC
2SC	2SC	2HC	2HC	2HC	2HC
2DC	2DC	2SC	2SC	2DC	2DC
PX	PX	PX	PX	PX	PX

in Figure 4 in logarithmic scale. Note that the most prominent changes in the frequency of corrections occurs in the abundance of 2SC corrections (orange line in Fig. 4), which is much larger in amplitude than the changes in the number of 2DC corrections (green line in Fig. 4). Consequently, it becomes evident that the permutations 1–4 perform similarly well, while the number of correction steps starts to increase for permutation 5 and becomes considerably higher for permutation 6. The lower performance of permutation 5 and specially that of permutation 6 is due to the early position of the 2SC corrections in the sequence of corrections (Table 1). When this algorithm is executed too early in the sequence of correction steps, it inserts a high number of 'corrected' pixels, which may cause false determination of edges during the skeletonisation of the network.

As mentioned above, permutations 1 to 4 perform similarly well. In order to compile with our criterion to minimise the number of correction steps, we use permutation 1, which fulfills this requirement.

3.4 Skeletonisation

The binarisation and the correction algorithms yield a binarised and 'repaired' network, whose veins have varying thickness. To extract the underlying graph, this extended network is collapsed to its line skeleton by applying the line thinning algorithm of Zhang & Suen [23]. This operation yields a network whose veins have the uniform thickness of a single pixel. Such networks may be conveniently analysed for the occurrence of nodes and the

lengths of the connecting edges (i.e. veins and venules) may easily be measured.

The skeletonised graph is once more subjected to a single-pixel diagonal correction (1DC). This operation is required to make any diagonal edge of the skeletonised graph accessible for the measurement of the length of the veins. The rationale for this procedure will be explained in section 3.5.2.

The skeletonised graph is compared with the background-subtracted images. A superposition of the extracted graph on the image of the plasmodial vein network shows a very good correspondence of the graph with the network (Fig. 2). In this figure, the recognised segments and nodes are plotted in green and red, respectively. A comparison between the extracted graph and its original, i.e. the plasmodial vein network, shows that the network is very successfully mapped on the graph. The latter is now available for analysis.

3.5 Determination of nodes and edges of the graph

The algorithms used to identify the nodes and edges operate as sliding windows on the matrix that contains the binarised and corrected image. The entries of this type of matrix are either 0 (black) or 255 (white). The search algorithms operate only in horizontal and vertical direction, i.e. operations along the diagonals are not performed.

3.5.1 Node detection

The matrix of the binarised and corrected images is scanned by a $5 \times 5 \ \text{pix}^2$ window and the neighbourhood of the central pixel is investigated. A node corresponds to any black pixel which has three or more black pixels in its $3 \times 3 \ \text{pix}^2$ vicinity. Whenever such a pixel is detected, its value in the matrix defining the binarised image is set to 50.

Due to the necessary application of the 1DC correction after binarisation, three types of nodes exist: in addition to nodes made of a single pixel, there are a few nodes comprising either 4 or 6 pixels. They arise when diagonal segments hit a node. The 1DC algorithm causes this node to be more extended. In order to avoid multiple counting, the algorithm first searches for the most extended (6-pixel) nodes, before it repeats the operation for the 4- and 1-pixel nodes. These 4 and 6 pixel nodes are also marked in the matrix by the value 50.

The connectivity of a node is found by summing up over the number of segments connected to the individual node; in other words, the number of black pixels surrounding the nodes are counted.

3.5.2 Edge detection and measurement of edge lengths

Starting from a detected node, the black pixels connected to this node are tracked. Again, this tracking is performed in horizontal and vertical directions only. The coordinates of the surrounding black pixel are marked, and the center of the $3 \times 3 \ \text{pix}^2$ search window is moved to this pixel. Now the operation is started again, searching for horizontal or vertical neighbouring black pixels. This operation is repeated as long as no other node is hit, or as a pixel is found, whose surrounding pixels are all white. In the former case, the edge connects two nodes, in the later the edge (or vein) terminates.

Once a segment is found, the pixels forming it are marked. The programme then returns to the starting node in order to track any additional edge departing from the node. Once all segments leaving from a node have been detected, the programme jumps to the next node.

To obtain an accurate count of the edges, the detected edges should be marked in a matrix. This labelling should prevent multiple counting of the edges.

Finally, the lengths of the edges are determined by counting the number of detected black pixels along every single edge (or venule).

4 Graph theoretical analysis of the *P. polycephalum* vein network

To gain a better understanding of the organisation of the plasmodial vein network of *P. polycephalum*, the characteristics of this network were analysed. The veins or venules form the edges of the graph and connect adjacent nodes. The graph **G** is given as

$$\mathbf{G} = \mathbf{W} \cdot \mathbf{A} \qquad (1)$$

where **A** is the adjacency matrix, while **W** is the matrix containing the lengths of the veins.

Interestingly, each node is always connected to exactly three other nodes, yielding a node degree k of 3 (Fig. 5). Graphs with delta distributed node degrees are called regular, and in the case of $k = 3$, cubic graphs. Regular graphs are characterised by a constant ratio of edges to nodes $M = \frac{1}{2}kN$, where M and N are the number of edges (veins) and nodes, respectively. This correlation is indeed observed for the *P. polycephalum* vein networks, where $M = (1.56 \pm 0.06)N$. The small deviation from the predicted value of $\frac{1}{2}k = 1.5$ is due to some venules that are ruptured into two disconnected segments by the process of coarsening of the vein network.
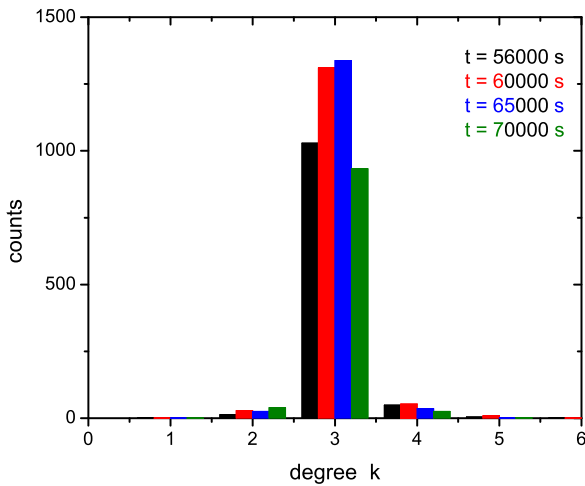
Figure 5 – Node degree k (or connectivity) of the individual nodes of the vein network of *P. polycephalum* as measured at different instants in time. Nodes have almost exclusively a node degree of $k = 3$.

We have also studied the distribution of the lengths of the veins between connected nodes. The lengths of the veins were counted in pixels, where 1 pixel corresponds to 0.08 mm. The distribution of the lengths of the vein segments follows the exponential dependence

$$P(x) = P_0 e^{-\gamma x} \tag{2}$$

where $P(x)$ is the probability to find a segment of length x in the network. The experimental data as well as the fit according to eq. 2, where $\gamma = 0.104 \pm 0.004$, is presented in Figure 6.
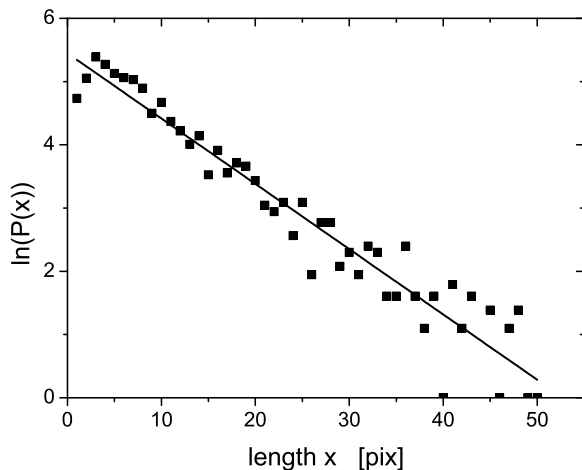

Figure 6 – Exponential scaling of the length distribution of vein segments of the network shown in Figure 2.

The analysis of the network structure of the plasmodial vein network of *P. polycephalum* reveals a constant node degree $k = 3$. Such a vein network forms a classical graph, namely a regular

graph. This graph is weighted by the different lengths of the vein segments, whose abundance shows a exponential distribution.

5   DISCUSSION

The present article reports on the detection and extraction of the vein network of the myxomycete *Physarum polycephalum* from data acquired in the form of images. In order to obtain an insight in the structure of the network formed by the veins and venules, an area has to be monitored, which is relatively large compared to the size and extension of the individual venules. This results in the challenging task of extracting tiny objects from the images.

We have presented a protocol that contains image processing operations and image correction algorithms (Fig. 1), which is able to extract the network of *P. polycephalum* from images. It has been found, that correction algorithms involving the correction of a single pixel should be performed prior to any algorithm that corrects more than one pixel at a time. If twin-pixel correction algorithms are performed too early in the sequence of corrections, a substantially larger number of individual corrections are executed. This leads to binarised graphs that contain areas (or islands) of black pixels, a situation which complicates or even jeopardises the graph theoretical analysis of the extracted graph.

In the special case of the plasmodium of *P. polycephalum*, the choice of a suitable threshold for the binarisation of the background subtracted image turns out to be important. This is due to the fact that the apical zone of the plasmodium, i.e. the domain exploring the surroundings of the cell (seen in the bottom of Fig. 2), has not yet established a vein network. It consists of an extended membrane that contains streaming protoplasm. In the binarised images the apical zone is seen as a domain of extended areas of black pixels. This area has to be excluded from the analysis, since any skeletonisation of this domain leads to the emergence of edges, whose positions do not necessarily correspond to the position of the cytoplasmic fingers.

Our preliminary analysis of the graph spanned by the veins and venules of *Physarum polycephalum* reveals the graph as being a regular graph. Furthermore, the evolving graph of *P. polycephalum* remains regular at all times, never forming any hubs. From a graph theoretical aspect, this contrasts to the vast majority of complex real world networks studied recently, which were shown to possess 'small-world' or 'scale-free' characteristics. These characteristics are completely absent in the vein network of *P. polycephalum*.

The regular graph of *P. polycephalum* venules is weighted by the lengths of the veins connecting the nodes. The length distri-

bution was found to be exponential. Clearly, there is also another weighting that needs to be examined in future, namely the weighting by the thickness of the veins. This should also provide some insights into the protoplasmic fluxes present in the vein network.

In addition to these first insights into the graph topology, our preliminary studies of the vein network of *P. polycephalum* provide a proof of concept that the extracted data can be used to characterise the graph properties of the vein network. A detailed analysis of the graph properties of this plasmodial vein network and of its evolution is in progress.

ACKNOWLEDGMENTS

REFERENCES

[1] WATTS DJ & STROGATZ SH. 1998. Collective dynamics of 'small world' networks. Nature, 393: 440–442.

[2] ALBERT R & BARABÁSI A-L. 2002. Statistical mechanics of complex networks. Rev. Mod. Phys., 74: 47–97.

[3] BOCCALETTI S, LATORA V, MORENO Y, CHAVEZ M & HWANG D-U. 2006. Complex networks: Structure and dynamics. Phys. Rep., 424: 175–308.

[4] BARRAT A, BARTHÉLEMY M, PASTOR-SATORRAS R & VESPIGNANI A. 2004. The architecture of complex weighted networks. Proc. Natl. Acad. Sci USA, 101: 3747–3752.

[5] GUIMERÀ R, MOSSA S, TURTSCHI A & AMARAL LAN. 2005. The worldwide air transportation network: Anomalous centrality, community structure, and cities' global roles. Proc. Natl. Acad. Sci USA, 102: 7794–7799.

[6] LI W & CAI X. 2007. Empirical analysis of a scale-free railway network in China. Physica A, 382: 693–703.

[7] DOMÉNECH A. 2009. A topological phase transition between small-world and fractal scaling in urban railway transportation networks? Physica A, 388: 4658–4668.

[8] FAGIOLO G, REYES J & SCHIAVO S. 2009. World-trade web: Topological properties, dynamics, and evolution. Phys. Rev. E, 79: 036115.

[9] BARABÁSI AL, JEONG H, NÉDA Z, RAVASZ E, SCHUBERT A & VICSEK T. 2002. Evolution of the social network of scientific collaborations. Physica A, 311: 590–614.

[10] LESS JR, SKALAK TC, SEVICK EM & JAIN RK. 1991. Microvascular architecture in a mammary carcinoma: Branching patterns and vessel dimensions. Cancer Res., 51: 265–273.

[11] BUHL J, GAUTRAIS J, SOLÉ RV, KUNTZ P, VALVERDE S, DENEUBOURG JL & THERAULAZ G. 2004. Efficiency and robustness in ant networks of galleries. Eur. Phys. J.B, 42: 123–129.

[12] BUHL J, GAUTRAIS J, DENEUBOURG JL, KUNTZ P & THERAULAZ G. 2006. The growth and form of tunneling networks in ants. J. Theor. Biol., 243: 287–298.

[13] WEARNE SL, RODRIGUEZ A, EHLENBERGER DB, ROCHER AB, HENDERSON SC & HOF PR. 2005. New techniques for imaging, digitization and analysis of three-dimensional neural morphology on multiple scales. Neurosci., 136: 661–680.

[14] MARWAN W. 2001. Photomovement and photomorphogenesis in *Physarum polycephalum*. Targeting of cytoskeleton and gene expression by light, in Häder D-P, Ledert M (eds.). Photomovements, Elsevier, Amsterdam, pp. 561–587.

[15] UEDA T. 2005. An inteligent slime mold: A self-organizing system of cell shape and information, in Armbruster D, Kaneko K, Mikhailov AS (eds.). Networks of interacting machines, World Scientific, New Jersey, pp. 221–255.

[16] ADAMATZKY A & JONES J. 2008. Towards *Physarum* robots: Computing and manipulating on water surface. J.Bionic Eng., 5: 348–357.

[17] NAKAGAKI T, YAMADA H & TÓTH Á. 2000. Maze-solving by an amoeboid organism. Nature, 407: 470.

[18] NAKAGAKI T, KOBAYASHI R, NISHIURA Y & UEDA T. 2004. Obtaining multiple separated food sources: Behavioural intelligence in *Physarum polycephalum*. Proc. R. Soc. Lond. B, 271: 2305–2310.

[19] NAKAGAKI T, IIMA M, UEDA T, NISHIURA Y, SAIGUSA T, TERO A, KOBAYASHI R & SHOWALTER K. 2007. Minimum-risk path finding by an adaptive amoebal network. Phys. Rev. Lett., 99: 068104.

[20] NAIB-MAJANI W, OSBORN M, WEBER K, WOLFARTH-BOTTERMANN K-E, HINSSEN H & STOCKEM W. 1983. Immunochemistry of the acellular slime mould *Physarum polycephalum*. J. Cell Sci., 60: 13–28.

[21] KAGAWA Y & TAKAMATSU A. 2009. Synchronisation and spatiotemporal patterns in coupled phase oscillators on a weighted planar network. Phys. Rev. E, 79: 046216.

[22] AKITAYA T, OHSAKA S, UEDA T & KOBATAKE Y. 1985. Oscillations in intracellular ATP, cAMP and cGTP concentration in relation to rhythmical sporulation under continuous light in the myxomycete *Physarum polycephalum*. J. Gen. Microbiol., 131: 195–200.

[23] ZHANG T & SUEN C. 1984. A fast parallel algorithm for thinning digital patterns. Comm. ACM, 27: 236–239.