Journal of
Risk and Financial Management

Article

Diffusion on the Peer-to-Peer Network

Julien Riposo

Independent Researcher, 59118 Wambrechies, France; julien.riposo@gmail.com

Abstract: In a peer-to-peer complex environment, information is permanently diffused. Such an environment can be modeled as a graph, where there are flows of information. The interest of such modeling is that (1) one can describe the exchanges through time from an initial state of the network, (2) the description can be used through the fit of a real-world case and to perform further forecasts, and (3) it can be used to trace information through time. In this paper, we review the methodology for describing diffusion processes on a network in the context of exchange of information in a crypto (Bitcoin) peer-to-peer network. Necessary definitions are posed, and the diffusion equation is derived by considering two different types of Laplacian operators. Equilibrium conditions are discussed, and analytical solutions are derived, particularly in the context of a directed graph, which constitutes the main innovation of this paper. Further innovations follow as the inclusion of boundary conditions, as well as the implementation of delay in the diffusion equation, followed by a discussion when doing approximations useful for the implementation. Numerous numerical simulations additionally illustrate the theory developed all along the paper. Specifically, we validated, through simple examples, the derived analytic solutions, and implemented them in more sophisticated graphs, e.g., the ring graph, particularly important in crypto peer-to-peer networks. As a conclusion for this article, we further developed a theory useful for fitting purposes in order to gain more information on its diffusivity, and through a modeling which the scientific community is aware of.

Keywords: blockchain; bitcoin; peer-to-peer; diffusion theory; matrix theory; graph theory; partial differential equations

Citation: Riposo, Julien. 2022. Diffusion on the Peer-to-Peer Network. *Journal of Risk and Financial Management* 15: 47. https://doi.org/10.3390/jrfm15020047

Academic Editor: Svetlozar Rachev

Received: 21 December 2021
Accepted: 14 January 2022
Published: 20 January 2022

1. Introduction

This document aims at describing the diffusion of information in the peer-to-peer (P2P) network related to a crypto, for instance Bitcoin, through a fundamental diffusion approach. The information could be anything, but the object of communication between agents (i.e., miners and users), and, to add context, it could be the *hash calculation*. This designates the processes which calculate hashes for mining blocks, given relevant remaining transactions to hash. More specifically, miners have the purpose of *hash* transactions within a list of transactions—the *memory pool*—which appear in the next block in the blockchain Lipton and Treccani (2021). Each transaction circulates from node to node in the memory pool used for remembering all new transactions, and miners receiving new transaction lists, supposing no double spending, start to generate hashes to create the new block. To this extent, the *hash calculation* starts as long as miners receive the list of transactions, thus the hash calculation can be seen as the information of transactions diffusing along the network (see Shahsavari et al. (2017) for a more theoretical discussion). The more a miner receives transactions to be mined, the more he/she is likely to calculate hashes, i.e., the stronger the hash calculation. On the contrary, if there is a low amount of information, i.e., not so many transactions to hash, then the lower the amount of transactions are to be hashed, and the lower the hash calculation. Conversely, if the hash calculation is high, this means that the miner has likely received a long list of transactions ready to be hashed. The hash calculation is intimately linked to the hash rate, so we will name this type of diffusion: *diffusion of hash rate*. In another more economical context, the diffused information could simply be Bitcoin cash flows.

This is true only if all miners have equal power of computing hashes, which we know is not realistic. This is not an issue: the diffusion model should depend on each node, and we will need to incorporate a *rate of creation* of hashes to assign to each node. In essence, not only do we have a diffusion of hash rate, but also the creation of hashes, increasing the hash rate for each miner. Being able to derive the hash rate with respect to time and node is being able to *predict* such an evolution for all nodes, at a given time and context, which is one main motivation for this approach.

A simple but fundamental physics model of diffusion is given by the famous *diffusion equation*,

$$\frac{\partial h}{\partial t} = D\Delta h + \Gamma, \tag{1}$$

where h is the hash rate (or more generally, a 'rate' of information), t is the time variable, D is the diffusion coefficient throughout the network, and Γ is the local hash creation rate. The operator Δ is called the *Laplacian* operator. One may think we could handle this approach and solve the equation for h. However, what do spatial coordinates represent? It seems that the majority of physics fields where this equation applies contains spatial coordinates, which are real numbers with respect to a pre-defined coordinate system. This is not the case for a proper network.

This means we need to handle the diffusion theory on a graph. There have already been established approaches somehow related to the diffusion description on a graph, with many applications (e.g., Bondy and Murty 2007; Pacreau et al. 2021; Thanou et al. 2017; Viriyasitavat and Hoonsopon 2019). We discuss some of the approaches for modeling diffusion on a network.

The paper An et al. (2021) specifies an optimality approach for minimizing the maximum regret due to decision taken in an uncertainty environment, and specifically calculates the maximal regret through a linear program. This program has the advantage of considering the connectivity of the network, and the evolution of the information diffusion should be a relevant complement to describe the structure of the network. The paper Mikhaylov (2021) defends Hayek's theory of private money, which, once applied to a state, should avoid a significant amount of private money within the population of this state and reduce the inequalities by means of further regulations through central banks digital currencies (CBDC). This imposes a certain structure in the network of exchanges: the current Bitcoin P2P network is distributed, and the application of this theory would make it change into a more centralized network. The diffusion properties may significantly change from one type to the other (see Section 7 for numerical illustrations), and our approach applies to both different types. The authors in the paper Hollanders et al. (2014) model a P2P network as a bipartite graph, where one family is the set of files, and the other is the set of peers. This way of modeling allows to find an 'S' shape in the dynamic of the information diffusion, and such a shape is a strong expectation of the social networks community. The equations guiding the evolution are through the susceptible/infectious (SI) equations. There are underlying assumptions, such as the strong one: the constancy of population. The paper Manini and Gribaudo (2008) engages a probabilistic approach of diffusion of files into a P2P network. In Musa et al. (2018), the diffusion methodologies applied to a P2P network are explicitly reviewed. If not based on the SI approach, the models are essentially probabilistic. Finally, the authors in Li et al. (2019) see the blockchain as a kind of Markov chain.

Interestingly, all the mentioned approaches overall focus on probabilistic modeling for the diffusion processes on a graph. This present study proposes to focus on the more fundamental aspects of the diffusion behavior, mainly governed by an equation of the style of Equation (1). It is worth stressing that the operator Γ is very general, and can be the source of randomness through, for instance, non-linearities, bringing chaos into the system. However, one possibly can also include probability in Γ, which would make the present study very general. More conceptually, Γ can gather any network specificity, such as its consensus, since it is related to sources of information.

However, some definitions already introduced in some of the previous studies, and including the Laplacian operator, may need to be further discussed, especially from a more intuitive and fundamental view of sharing information between two vertices of a graph. This is especially true when the graph is directed. Thus, in the context of diffusion on a P2P network, we need, for deriving the appropriate diffusion equation, to introduce the derivative and the integration on a graph, for a function f defined on each node x at a given time t. The Laplacian operator will follow. One understands that deriving the appropriate diffusion equation is a task in itself. Then, once derived, we need to solve the equation, which represents another task. In addition, we will see that there are different possible Laplacian operators for a diffusion theory (at least two), and we will focus on them separately. We will need some linear algebra tools to do it, as a graph is equivalent to its adjacency matrix, and, conversely, any matrix gives a (weighted) graph.

We now discuss the two hypotheses made (and mitigated) in this study. The first one is that graphs are undirectional. This is a strong assumption, as it means that once a vertex agent receives information that it did not previously have, then it will give back part of it to the source. Although this might be convenient in the context of bitcoin exchange (Alice sends some bitcoins to Bob, but also to herself, i.e., she generates new addresses she owns to send part of the total amount of bitcoins), a piece of information, whatever its type, has no reason to come back to its source. Considering that a directed graph is essential if we need to describe the interactions within a P2P network, this is what we do in this paper, and this constitutes the main innovation of the present study.

Another hypothesis is made in this modeling: when a vertex agent receives a piece of information, he/she immediately treats it and diffuses it to its neighbors. In other words, the response to the information reception is immediate. This is also a strong assumption, as, for instance, cash flows are performed in a delayed manner. Thus, Alice sends some bitcoins to Bob, but Bob does not immediately send a part (or more) of it to Charlie. In fact, the immediate response is the straightforward consequence of the structure of the diffusion equation, whose solutions are a differentiable function of time. It turns out that one way to make a non-differentiable solution at some given times is to introduce time dependency in the Laplacian operator, and the Heaviside function (whose value is 0 below a given time, and 1 above it) is a relevant candidate for differentiability breaking at selected points in time. Introducing Heaviside functions in the Laplacian operator is a much easier task in the context of graph modeling, for the reason that there are no discontinuity issues for the variable x, usually continuous, and here being discrete (and representing the node index). We engage this point of view at the end of the study.

The rest of this paper is depicted as follows. Section 2 is an introduction to the modeling, with many fundamental definitions which will imply the Laplacian ones, as well as the diffusion theory. Section 3 derives and solves the diffusion equation defined on an undirected graph. It is worth pointing out that the two previous sections are strongly inspired from Chung et al. (2007). The rest of this paper constitutes the main innovations. Section 4 introduces two different types of equilibrium notion, the standard one ($t \to \infty$) and the blockchain one ($t = 10$ min for Bitcoin), as well as some conceptual links between both. Section 5 shows the implementation of the derived solution in Section 3. Section 6 is an important generalization of the approach done thus far: it introduces the diffusion equation and its resolution in a directed graph, very important properties of the P2P network as already discussed above. Section 7 shows numerical illustrations of the previous section for several famous networks (including the Erdos–Reyni network, or the binary tree). Section 8 is the direct implication of the previous directed graph case when there are boundary conditions, typically certain vertices constantly sending information to the rest of the network. Finally, Section 9 includes delay in the response of nodes when they receive information. The last section concludes the study.

2. Modelling the P2P Network

2.1. Introduction to Graph Theory

Let $G = (E, V)$ be a graph, that is a set of nodes E and a set of hedges V, composed of pairs of nodes in E. Thus, if $x \in V$, i.e., x is a node of G, and $y \in E$ is another node of G; if $(x, y) \in V$, we say that the nodes x and y are connected in G. We suppose that G is connected (that is, there exists a connection between any two nodes of G), unoriented (x to y or y to x is the same) and unweighted (all the paths are equally effective); this last hypothesis may be relaxed in the future. An example of such a graph is shown in Figure 1.

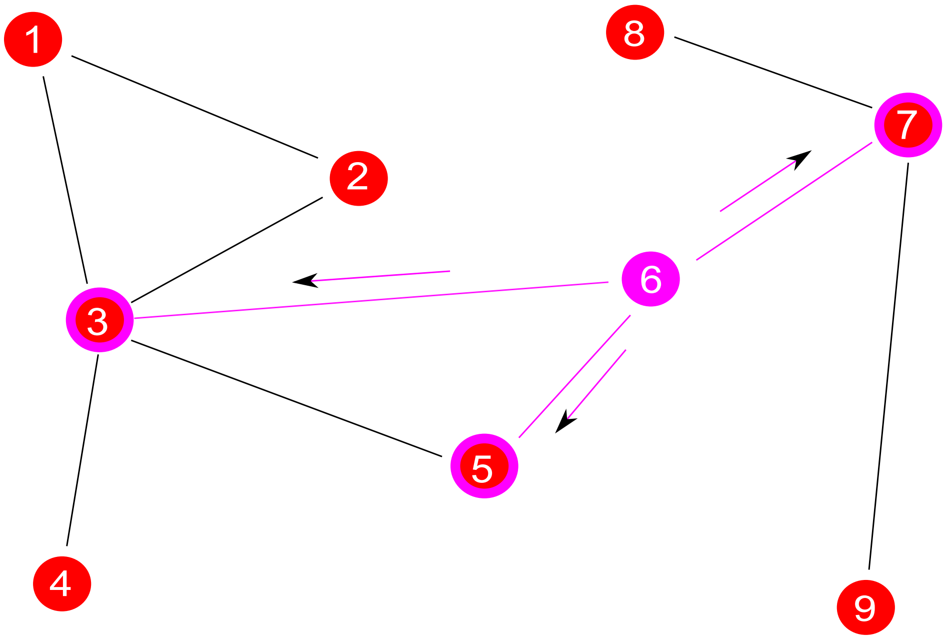

Figure 1. Illustration of a graph. The balls are nodes, which are connected by edges, the segments. Here, nodes have between one and four neighbors. In particular, node 6 diffuses information (e.g., list of transactions) to its neighbors, i.e., nodes 3, 5, and 7.

There is a matrix equivalence for such a graph. Node 1 is connected to nodes 2 and 3, in Figure 1, but not to any other node. We can represent the graph G by a matrix A, which is named the *adjacency matrix*. If we arbitrarily label nodes with numbers, the adjacency matrix for the graph G in Figure 1 is given by

$$A = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}. \quad (2)$$

The first row corresponds to the connections of node 1. Node 1 is connected to nodes 2 and 3 only, so columns 2 and 3 for the first row prints 1; otherwise, it prints 0, and so on. The diagonal (from top left to bottom right) of this matrix is composed of zeroes. It actually turns out the this matrix is symmetric (i.e., with respect to its diagonal). In this example, E

is isomorphic to $\{1, 2, \ldots, 9\}$—we will identify both sets if there is no ambiguity—and the element a_{ij} of A is 1 if $(i, j) \in V$ for $(i, j) \in E^2$.

Studying the matrix A or the graph G is therefore equivalent, and the P2P network could indeed be represented this way. In addition, in Figure 1, node 6 spreads the information of new transactions to nodes 3, 5, and 7. Node 6 starts to hash first, and then nodes 3, 5, and 7 start once they receive the information: the 'hash rate' propagates. Here, node 6 firstly computes hashes (purple), and secondly nodes 3, 5, and 7 compute hashes (they become purple). This approach generalizes to all the nodes that *create* hashes, i.e., diffuses list of transactions to other nodes as well as starting to compute hashes, and therefore will make its neighbors compute hashes.

The arising question is, how is the diffusive process of such hash information, i.e., can we describe it and thus allow a prediction? The answer is yes.

2.2. Derivative

We need to introduce the derivative and the integration properly before doing anything else. It is worth noting that significant effort has been made to get closer to the notations involved in the differential calculus. From now on, the sequence of this paper will be mathematical, unless specified otherwise. The diffusive quantity of interest, here the hash rate, is given by the function f, function of nodes $x \in V$ and time $t \in \mathbb{R}_+$.

In the following, we focus on a graph $G = (E, V)$ with associated adjacency matrix $A = (a_{ij})_{1 \leq i,j \leq n}$ ($n \in \mathbb{N}^* \backslash \{1\}$ nodes). The graph G is connected, unoriented and unweighted so that the matrix A is symmetrical, and its elements are in $\{0, 1\}$. We finally call $\mathcal{N}(x)$ the set of all neighbors of $x \in V$, and we write $\mathrm{d}_A x = \mathrm{Card}\, \mathcal{N}(x)$ the degree of x, i.e., number of neighbors of x, and we call $\mathrm{Vol}(G) = \sum_{x \in V} \mathrm{d}_A x$ the volume of the graph G. We write $x \in V$ or $x \in G$ equivalently in the sequence.

Definition 1 (Derivative)**.** *Let f be a function of $E \times \mathbb{R}_+$ in $\mathbb{R}$.*[1] *The directional derivative of f from $z \in G$ to $y \in G$ is given by*

$$\partial_z f(y) = (f(y) - f(z))\, a_{yz}. \tag{3}$$

The operator ∂_z is a linear map of f, for any $z \in G$.

Figure 2 represents the derivative. The rest of the definitions depend on this, so it is important to well understand this last one.

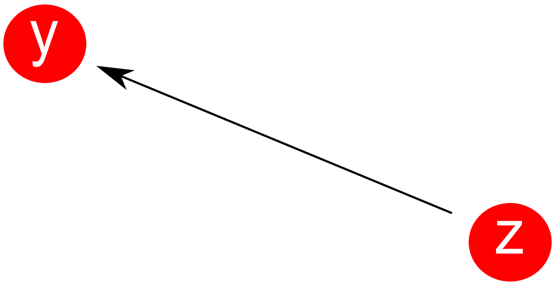

Figure 2. The directional derivative of f from z to y. This actually represents the (finite) variation of f for the information going from node z to node y. It is positive if y gets more information than z, and negative otherwise. This concept makes sense only if y and z are neighbors; hence, the presence of $a_{yz} \in \{0, 1\}$ in the definition.

Definition 2 (Gradient)**.** *Let $f : E \times \mathbb{R}_+ \to \mathbb{R}$. The gradient of f at $y \in G$ is the vector given by* [2]

$$\nabla^y f = (\partial_z f(y))_{z \in \mathcal{N}(y)} = \begin{pmatrix} \partial_1 f(y) \\ \partial_2 f(y) \\ \vdots \\ \partial_{\mathrm{d}_A y} f(y) \end{pmatrix}. \tag{4}$$

The gradient at a node is the collection of directional derivatives toward this node.

Definition 3 (Laplacians)**.** *Let $f : E \times \mathbb{R}_+ \to \mathbb{R}$. The normalized Laplacian of f from $x \in G$ to $y \in G$ is given by*

$$\Delta_x f(y) = \frac{1}{\mathrm{d}_A y} \sum_{z \in \mathcal{N}(x)} \partial_z \partial_z f(y), \tag{5}$$

while the non-normalized Laplacian of f from $x \in G$ to $y \in G$ is given by

$$L_x f(y) = \sum_{z \in \mathcal{N}(x)} \partial_z \partial_z f(y). \tag{6}$$

The global normalized Laplacian of f at x is given by

$$\Delta f(x) = \frac{1}{\mathrm{d}_A x} \sum_{z \in G} \partial_z \partial_z f(x), \tag{7}$$

while the global non-normalized Laplacian of f at x is given by

$$Lf(x) = \sum_{z \in G} \partial_z \partial_z f(x). \tag{8}$$

Lf (or Δf) is said to be the Laplacian of f.

The Laplacian Δ of f is the average of the second directional derivative of f at node x, while the Laplacian L of f is the average of the second directional derivative of f at node x. We then could define the following operator:

Definition 4 (Divergence)**.** *Let $x \in G$ and $f_z : E \times \mathbb{R}_+ \to \mathbb{R}$, $z \in \mathcal{N}(x)$ be $\mathrm{d}_A x$ functions. Define the vectorial function $f = (f_z)_{z \in \mathcal{N}(x)}$. The divergence of f at $x \in G$ is given by*

$$\nabla^x \cdot f = \sum_{z \in \mathcal{N}(x)} f_z(z), \tag{9}$$

or

$$\nabla^x \cdot f = \frac{1}{\mathrm{d}_A x} \sum_{z \in \mathcal{N}(x)} f_z(z), \tag{10}$$

depending on whether we choose L or Δ.

This is the sum (or average) value of f for a given node, the average being taken at each of the considered node's neighbors.

Proposition 1 (Div of grad is Laplacian)**.** *For any $x \in G$, we have*

$$\nabla^x \cdot \nabla^x f = L_x f \text{ or } \Delta_x f, \tag{11}$$

depending on whether we choose L_x or Δ_x for the definition of the divergence.

Proof. Note that, for all $y, z \in G$, we have

$$\begin{aligned}\partial_z\partial_z f(y) &= \partial_z((f(y) - f(z))\, a_{yz}) \\ &= (\partial_z f(y) - \partial_z f(z))\, a_{yz} \\ &= \big((f(y) - f(z))\, a_{yz} - 0\big)\, a_{yz} \\ &= (f(y) - f(z))\, a_{yz}^2.\end{aligned}$$

Since $a_{yz}^2 = a_{yz} \in \{0, 1\}$, we therefore have that $\partial_z\partial_z f(y) = \partial_z f(y)$. It turns out that $\nabla^x \cdot \nabla^x f = \Delta_x f$, or $\nabla \cdot \nabla = \Delta$. □

Thus, we note that $\partial_z\partial_z = \partial_z$, and it turns out that $\Delta f(x) = \Delta_x f(x)$, for any $x \in G$. A straightforward but essential property for the Laplacian is depicted as follows.

Proposition 2. *Let $f : E \times \mathbb{R}_+ \to \mathbb{R}$. The Laplacians of f at $x \in G$ satisfy the following property:*

$$Lf(x) = \mathrm{d}_A x\, f(x) - \sum_{z \in \mathcal{N}(x)} f(z), \tag{12}$$

$$\Delta f(x) = f(x) - \frac{1}{\mathrm{d}_A x} \sum_{z \in \mathcal{N}(x)} f(z). \tag{13}$$

Thus, the Laplacian of f at x is the value of f at x, minus the average value of f taken by all its neighbors.

Proof. From the definition, and since $a_{xz} = 1$ if and only if $z \in \mathcal{N}(x)$, we have

$$\Delta f(x) = \frac{1}{\mathrm{d}_A x} \sum_{z \in \mathcal{N}(x)} \partial_z\partial_z f(x) = \frac{1}{\mathrm{d}_A x} \left(\sum_{z \in \mathcal{N}(x)} f(x) - \sum_{z \in \mathcal{N}(x)} f(z) \right),$$

which implies the results. □

This actually allows to introduce the Laplacian matrix.

Definition 5 (Laplacian matrix)**.** *The non-normalized Laplacian matrix is given by*

$$L = \mathcal{D} - A, \tag{14}$$

where A is the adjacency matrix, and $\mathcal{D} = \operatorname{diag}(\mathrm{d}_A x, \mathrm{d}_A y, \ldots)$ is the diagonal matrix of degrees.

The Laplacian matrix is given by

$$\Delta = \mathcal{D}^{-1} L = 1 - \mathcal{D}^{-1} A, \tag{15}$$

where 1 is the identity matrix.

Contrary to any other field in mathematics, there are no criteria of derivability. The function just needs to be defined at the two considered points.

2.3. Integration

After having defined the derivatives, we have to define the integration.

Definition 6. *Let $f, h : E \to \mathbb{R}$. We introduce the map scalar product of f by h on any sub-graph $g \subseteq G$ of G, as follows:*

$$\langle f, h \rangle_g = \sum_{x \in g} f(x)\, h(x). \tag{16}$$

The following proposition is straightforward, but fundamental.

Proposition 3 (Scalar product)**.** *The map $\langle\cdot,\cdot\rangle_g : f,h \mapsto \langle f,h\rangle_g$ is a bilinear form from E^2 to $\mathbb{R}$, for any $g \subseteq G$. This justifies its name scalar product. Thus, the family $(G, \langle\cdot,\cdot\rangle)$ is said to be a Euclidean graph.*

In the following, the graph G is always considered a Euclidean graph.

Definition 7 (Integration)**.** *Let $f : E \to \mathbb{R}$. We introduce the integration of f on any sub-graph $g \subseteq G$ of G, as follows:*

$$\int_g f\,\mathrm{d}_A = \langle f, \mathrm{d}_A\rangle = \sum_{x\in g} f(x)\,d_A x. \tag{17}$$

d_A *is a measure on G by $\langle\cdot,\cdot\rangle$.*

Here again, the function, to be integrable in g, just needs to be defined on g. The previous and following properties are still satisfied for a function $f : E \times \mathbb{R}_+ \to \mathbb{R}$.

The above definition actually is consistent with integration by parts.

Proposition 4 (Integration by parts)**.** *Let $f,h : E \to \mathbb{R}$. We have*

$$\int_{\mathcal{N}(x)} h\,\Delta_x f\,\mathrm{d}_A = \int_{\mathcal{N}(x)} \Delta_x h\,f\,\mathrm{d}_A, \tag{18}$$

$$\int_G h\,\Delta f\,\mathrm{d}_A = \int_G \Delta h\,f\,\mathrm{d}_A. \tag{19}$$

Proof. We calculate

$$\begin{aligned}\int_{\mathcal{N}(x)} h\,\Delta_x f\,\mathrm{d}_A &= \sum_{y\in\mathcal{N}(x)} h(y)\,\Delta_x f(y)\,\mathrm{d}_A y = \sum_{y\in\mathcal{N}(x)}\sum_{z\in\mathcal{N}(x)} h(y)\,(f(y)-f(z))\\ &= \frac{1}{2}\sum_{y\in\mathcal{N}(x)}\sum_{z\in\mathcal{N}(x)} h(y)\,(f(y)-f(z)) + \frac{1}{2}\sum_{y\in\mathcal{N}(x)}\sum_{z\in\mathcal{N}(x)} h(z)\,(f(z)-f(y))\\ &= \frac{1}{2}\sum_{y\in\mathcal{N}(x)}\sum_{z\in\mathcal{N}(x)} (h(y)-h(z))\,(f(y)-f(z))\\ &= \sum_{y\in\mathcal{N}(x)}\sum_{z\in\mathcal{N}(x)} (h(y)-h(z))\,f(y) = \sum_{y\in\mathcal{N}(x)} \Delta_x h(y)\,f(y)\,\mathrm{d}_A y\\ &= \int_{\mathcal{N}(x)} \Delta_x h\,f\,\mathrm{d}_A.\end{aligned}$$

Equation (18) is proven. In addition, we have

$$\begin{aligned}\int_G h\,\Delta f\,\mathrm{d}_A &= \sum_{x\in G} h(x)\,\Delta f(x)\,\mathrm{d}_A x\\ &= \sum_{x\in G}\sum_{y\in G} h(x)\,(f(x)-f(y))\,a_{xy}\\ &= \frac{1}{2}\sum_{x\in G}\sum_{y\in G}(h(x)-h(y))\,(f(x)-f(y))\,a_{xy},\end{aligned}$$

which aims at proving Equation (19) as was previously done. □

We now have all the tools to develop the diffusion theory on a network, which is the object of the next session.

3. The Diffusion Equation

3.1. Derivation of the Equation

In order to derive the diffusion equation, we need the following essential proposition. In the sequence, we call $f : E \times \mathbb{R}_+ \to \mathbb{R}$ the number of hashes produced per miner, as a function of nodes and time.

Proposition 5. *The density of the current of hashes from at $x \in G$ at time t is given by the following law of diffusion:*

$$j(t, x) = -D\, \nabla^x f(t, \cdot), \tag{20}$$

where $D > 0$ is the diffusion coefficient of information.

We indeed assume that D is constant and is a network characteristic. Despite the fact that this law is very intuitive and is a typical *Fick* law (the information goes toward where there is not any information yet and hence the negative sign), further developments need to be made to rigorously justify this equation in the context of the network. Let us write $j(t, x) = (j_z(t, x))_{z \in \mathcal{N}(x)}$ the vector j at x and t. Thus, $j_z(t, x)$ represents the *flow* of information from z to x.

Here is the first important result of the paper.

Theorem 1. *Let $f : E \times \mathbb{R}_+ \to \mathbb{R}$ represent the information function. If D is the diffusion coefficient and $\Gamma : E \times \mathbb{R}_+ \to \mathbb{R}$ the creation function of the information, the following diffusion equation holds for any $x \in V$ and $t \in \mathbb{R}_+$:*

$$\frac{\partial f}{\partial t}(t, x) = -D\, Lf(t, x) + \Gamma(t, x) \tag{21}$$

Separately, the following diffusion equation also is valid

$$\frac{\partial f}{\partial t}(t, x) = -D\, \Delta f(t, x) + \Gamma(t, x) \tag{22}$$

As long as one remembers that it is local, these equations can simply be written as

$$\frac{\partial f}{\partial t} = -D\, Lf + \Gamma, \qquad \frac{\partial f}{\partial t} = -D\, \Delta f + \Gamma.$$

The minus sign is actually a necessity for the system to be described as diffusive, and not an explosive one, contrary to standard math and physics systems.

Proof. On the one hand, let δN be the quantity of hashes produced during the infinitesimal variation of time $\mathrm{d}t$, at node $x \in G$. The hash rate is the flux of rates given by $\phi = \delta N / \mathrm{d}t$, and the number of such hashes produced per neighbors, for node $x \in G$, from time t and $t + \mathrm{d}t$, is $f(t + \mathrm{d}t, x) - f(t, x) = \delta N / \mathrm{d}_A x$. It turns out that

$$\delta N = \frac{\partial f}{\partial t}(t, x)\, \mathrm{d}t\, \mathrm{d}_A x.$$

On the other hand, node x (i) receives the diffused information from its neighbor, implying δN_{enter} hashes, and (ii) can create (or destruct) information, implying δN_{create} hashes. We therefore must have the following *conservation of information*:

$$\delta N = \delta N_{\text{entering}} + \delta N_{\text{create}}. \tag{23}$$

On point (i), the quantity of information entering into node x is provided from all its neighbors, and thus $\delta N_{\text{entering}} = \sum_{z \in \mathcal{N}(x)} j_z(t, x)\, \mathrm{d}t$. On point (ii), we introduce the creation rate per node $\Gamma(t,x)$ at node x and time t, which is the rate per node per second hashes are calculated. We therefore have $\delta N_{\text{create}} = \Gamma(t,x)\,\mathrm{d}t\,\mathrm{d}_A x$.

Bearing all the above in mind, Equation (23) becomes

$$\frac{\partial f}{\partial t}(t,x)\,\mathrm{d}t\,\mathrm{d}_A x = \sum_{z\in\mathcal{N}(x)} j_z(t,x)\,\mathrm{d}t + \Gamma(t,x)\,\mathrm{d}t\,\mathrm{d}_A x,$$

Equivalently,

$$\frac{\partial f}{\partial t}(t,x) = \frac{1}{\mathrm{d}_A x} \sum_{z\in\mathcal{N}(x)} j_z(t,x) + \Gamma(t,x),$$

or

$$\frac{\partial f}{\partial t}(t,x) = \nabla^x \cdot j(t,x) + \Gamma(t,x).$$

According to Proposition 5, we have $\nabla^x \cdot j(t,x) = -D\,\Delta f(t,x)$ or $L\Delta f(t,x)$, which does conclude the proof. □

3.2. Laplacian Matrix: Eigenvalues/Eigenvectors

3.2.1. Quick reminder: Eigenvalues/Eigenvectors

Let $G = (E,V)$ be a graph, with $n \in \mathbb{N}$ nodes and the adjacency matrix A, and $F : E^n \to E^n$. We say that $\lambda \in \mathbb{C}$ is an *eigenvalue* of F if the function $F - \lambda Id$ (Id is the identity map) is not injective, which means that there exists a vector $V \in \mathbb{R}^n$, named *eigenvector*, such that $F(V) = \lambda V$, or component-wise $F(V(x)) = \lambda V(x)$ for $x \in G$. The function F has n eigenvalues, which we write λ_1, λ_2, . . ., λ_n, the index being chosen such that $|\lambda_1| \geq |\lambda_2| \geq \ldots \geq |\lambda_n| \geq 0$. To each eigenvalue λ_i, we associate its eigenvector V_i, for all $\{1,2,\ldots,n\}$.

Separately, the function F can be seen as a matrix M, and the above equation is thus written as $MV = \lambda V$, or component-wise $MV(x) = \lambda V(x)$. We adapt to the matrix viewpoint in the following. If M is symmetric, then it is diagonalizable and $(V_i)_{i\in\{1,2,\ldots,n\}}$ is an orthonormal basis of $\mathbb{R}^n$, that is

$$\langle V_i, V_j\rangle_G = \sum_{x\in G} V_i(x)\,V_j(x) = \delta_{ij}, \tag{24}$$

where δ_{ij} is the Kronecker symbol.

3.2.2. Laplacian: Eigenvalues/Eigenvectors

The above applies to the Laplacian matrix (see Definition 5). For instance, the non-normalized Laplacian matrix for Figure 1 writes as

$$L = \begin{pmatrix} 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 1 & -1 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & -1 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \end{pmatrix}. \tag{25}$$

The matrix L is symmetric and positive (semi-)definite. We write $(\mu_i)_{i\in\{1,2,\ldots,n\}}$ the eigenvalues of the matrix L such that $\mu_1 \geq \mu_2 \geq \ldots \geq \mu_n \geq 0$ with the associated eigenvalues $(v_i)_{i\in\{1,2,\ldots,n\}}$, forming an orthonormal basis of $\mathbb{R}^n$. The normalized Laplacian matrix writes

$$\Delta = \begin{pmatrix} 1 & -1/2 & -1/2 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1/2 & 1 & -1/2 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1/5 & -1/5 & 1 & -1/5 & -1/5 & -1/5 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1/2 & 0 & 1 & -1/2 & 0 & 0 & 0 \\ 0 & 0 & -1/3 & 0 & -1/3 & 1 & -1/3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1/3 & 1 & -1/3 & -1/3 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \end{pmatrix}. \tag{26}$$

The matrix Δ is only positive semi-definite.

In the sequence, we focus our attention on the matrix $\tilde{\Delta} = \mathcal{D}^{1/2}\,\Delta\,\mathcal{D}^{-1/2}$. This matrix is positive semi-definite, but also, since $\tilde{\Delta} = \mathcal{D}^{1/2}\,\mathcal{D}^{-1}\,L\,\mathcal{D}^{-1/2} = \mathcal{D}^{-1/2}\,L\,\mathcal{D}^{-1/2}$, it is symmetric. As an example, we have

$$\tilde{\Delta} = \begin{pmatrix} 1 & -1/\sqrt{2} & -1/\sqrt{10} & 0 & 0 & 0 & 0 & 0 & 0 \\ -1/\sqrt{2} & 1 & -1/\sqrt{10} & 0 & 0 & 0 & 0 & 0 & 0 \\ -1/\sqrt{10} & -1/\sqrt{10} & 1 & -1/\sqrt{5} & -1/\sqrt{10} & -1/\sqrt{15} & 0 & 0 & 0 \\ 0 & 0 & -1/\sqrt{5} & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1/\sqrt{10} & 0 & 1 & -1/\sqrt{6} & 0 & 0 & 0 \\ 0 & 0 & -1/\sqrt{15} & 0 & -1/\sqrt{6} & 1 & -1/3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1/3 & 1 & -1/\sqrt{3} & -/\sqrt{3} \\ 0 & 0 & 0 & 0 & 0 & 0 & -1/\sqrt{3} & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1/\sqrt{3} & 0 & 1 \end{pmatrix}. \tag{27}$$

We write $(\lambda_i)_{i\in\{1,2,\ldots,n\}}$ the eigenvalues of the matrix $\tilde{\Delta}$ such that $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n \geq 0$ with the associated eigenvalues $(V_i)_{i\in\{1,2,\ldots,n\}}$, forming an orthonormal basis of $\mathbb{R}^n$. The graph G is fully connected; therefore $\lambda_n = 0$.

It is interesting to further develop the eigenvector V_n associated with the eigenvalue $\lambda_n = 0$. Set v_n as the eigenvector of L associated with the eigenvalue $\lambda_n = \mu_n = 0$ (same common eigenvalue between L and $\tilde{\Delta}$). In addition, we have $Lv_n = 0$; hence, $\mathcal{D}v_n = Av_n$, which implies that $v_n = e/\sqrt{n}$ with the appropriate normalization.

Actually, we easily can establish that

$$\tilde{\Delta}\,\mathcal{D}^{1/2}\,v_n = \lambda_n \mathcal{D}^{-1}\,\mathcal{D}^{1/2}\,v_n = 0 = \lambda_n \mathcal{D}^{1/2}\,v_n.$$

It turns out that $V_n = \mathcal{D}^{1/2}\,v_n$, and with the appropriate normalization, we finally have

$$V_n(x) = \sqrt{\frac{\mathrm{d}_A x}{\mathrm{Vol}(G)}}. \tag{28}$$

Two additional points to bear in mind are that $\tilde{\Delta}$ is equivalent to Δ by construction; therefore, both matrices have the same eigenvalues. Finally, since $\tilde{\Delta} = \mathcal{D}^{-1/2}\,L\,\mathcal{D}^{-1/2}$, then, according to Sylvester's inertia theorem, $\tilde{\Delta}$ and L, and thus Δ, have the same number of negative, zero, and positive eigenvalues. Further developments can be done here.

3.3. Resolution of the Equation

We need the following lemma

Lemma 1. *The operator L is self-adjoint, in the sense that*

$$\langle h, L_x f\rangle_{\mathcal{N}(x)} = \langle L_x h, f\rangle_{\mathcal{N}(x)}, \tag{29}$$

$$\langle h, Lf \rangle_G = \langle Lh, f \rangle_G. \tag{30}$$

In addition, the operator $\tilde{\Delta}$ also is self-adjoint

$$\langle h, \tilde{\Delta}_x f \rangle_{\mathcal{N}(x)} = \langle \tilde{\Delta}_x h, f \rangle_{\mathcal{N}(x)}, \tag{31}$$

$$\langle h, \tilde{\Delta} f \rangle_G = \langle \tilde{\Delta} h, f \rangle_G. \tag{32}$$

Proof. The first two equations have proofs very similar to the one for Proposition 4. We thus focus on $\tilde{\Delta}_x$ and $\tilde{\Delta}$.

First we note, for any $x \in G$, that

$$\tilde{\Delta} f(x) = f(x) - \frac{1}{\sqrt{\mathrm{d}_A x}} \sum_{z \in \mathcal{N}(x)} \frac{f(z)}{\sqrt{\mathrm{d}_A z}} = \frac{1}{\sqrt{\mathrm{d}_A x}} \sum_{z \in \mathcal{N}(x)} \left(\frac{f(x)}{\sqrt{\mathrm{d}_A x}} - \frac{f(z)}{\sqrt{\mathrm{d}_A z}} \right).$$

It turns out that

$$\tilde{\Delta}_x f(y) = \frac{1}{\sqrt{\mathrm{d}_A y}} \sum_{z \in \mathcal{N}(x)} \left(\frac{f(y)}{\sqrt{\mathrm{d}_A y}} - \frac{f(z)}{\sqrt{\mathrm{d}_A z}} \right) a_{yz}, \quad \forall y \in G.$$

Bearing this in mind, we calculate

$$\begin{aligned}
\langle h, \tilde{\Delta}_x f \rangle_{\mathcal{N}(x)} &= \sum_{y \in \mathcal{N}(x)} h(y)\, \tilde{\Delta}_x f(y) = \sum_{y \in \mathcal{N}(x)} h(y) \frac{1}{\sqrt{\mathrm{d}_A y}} \sum_{z \in \mathcal{N}(x)} \left(\frac{f(y)}{\sqrt{\mathrm{d}_A y}} - \frac{f(z)}{\sqrt{\mathrm{d}_A z}} \right) a_{yz} \\
&= \sum_{y \in \mathcal{N}(x)} \sum_{z \in \mathcal{N}(x)} \frac{h(y)}{\sqrt{\mathrm{d}_A y}} \left(\frac{f(y)}{\sqrt{\mathrm{d}_A y}} - \frac{f(z)}{\sqrt{\mathrm{d}_A z}} \right) a_{yz} \\
&= \frac{1}{2} \sum_{y \in \mathcal{N}(x)} \sum_{z \in \mathcal{N}(x)} \frac{h(y)}{\sqrt{\mathrm{d}_A y}} \left(\frac{f(y)}{\sqrt{\mathrm{d}_A y}} - \frac{f(z)}{\sqrt{\mathrm{d}_A z}} \right) a_{yz} \\
&\quad + \frac{1}{2} \sum_{y \in \mathcal{N}(x)} \sum_{z \in \mathcal{N}(x)} \frac{h(z)}{\sqrt{\mathrm{d}_A z}} \left(\frac{f(z)}{\sqrt{\mathrm{d}_A z}} - \frac{f(y)}{\sqrt{\mathrm{d}_A y}} \right) a_{zy} \\
&= \frac{1}{2} \sum_{y \in \mathcal{N}(x)} \sum_{z \in \mathcal{N}(x)} \left(\left(\frac{h(y)}{\sqrt{\mathrm{d}_A y}} - \frac{h(z)}{\sqrt{\mathrm{d}_A z}} \right) a_{yz} \times \left(\frac{f(y)}{\sqrt{\mathrm{d}_A y}} - \frac{f(z)}{\sqrt{\mathrm{d}_A z}} \right) a_{yz} \right) \\
&= \sum_{y \in \mathcal{N}(x)} \sum_{z \in \mathcal{N}(x)} \left(\frac{h(y)}{\sqrt{\mathrm{d}_A y}} - \frac{h(z)}{\sqrt{\mathrm{d}_A z}} \right) a_{yz} \frac{f(y)}{\sqrt{\mathrm{d}_A y}} \\
&= \sum_{y \in \mathcal{N}(x)} \tilde{\Delta}_x h(y)\, f(y) = \langle \tilde{\Delta}_x h, f \rangle_{\mathcal{N}(x)}.
\end{aligned}$$

Equation (31) is proven. For Equation (32), we have

$$\begin{aligned}
\langle h, \tilde{\Delta} f \rangle_G &= \sum_{x \in G} h(x)\, \tilde{\Delta} f(x) = \sum_{x \in G} h(x) \frac{1}{\sqrt{\mathrm{d}_A x}} \sum_{y \in G} \left(\frac{f(x)}{\sqrt{\mathrm{d}_A x}} - \frac{f(y)}{\sqrt{\mathrm{d}_A y}} \right) a_{xy} \\
&= \frac{1}{2} \sum_{x \in G} \sum_{y \in G} \left(\left(\frac{h(x)}{\sqrt{\mathrm{d}_A x}} - \frac{h(y)}{\sqrt{\mathrm{d}_A y}} \right) a_{xy} \times \left(\frac{f(x)}{\sqrt{\mathrm{d}_A x}} - \frac{f(y)}{\sqrt{\mathrm{d}_A y}} \right) a_{xy} \right),
\end{aligned}$$

which aims at proving the result, as previously done. □

Here is the second important result of the paper. We have the following analytical expression for the hash rate per node.

Theorem 2. *The solution for the diffusion Equation (21) is given by*

$$f(t,x) = \sum_{i=1}^{n} \left(C_i^L \, \mathrm{e}^{-D\mu_i t} + \int_0^t \mathrm{e}^{-D\mu_i (t-\tau)} \langle \Gamma(\tau,\cdot), v_i \rangle_G \, \mathrm{d}\tau \right) v_i(x) \tag{33}$$

with $(C_1^L, C_2^L, \dots, C_n^L) \in \mathbb{R}^n$.

The solution for the diffusion Equation (22) is given by

$$f(t,x) = \frac{1}{\sqrt{\mathrm{d}_A x}} \sum_{i=1}^{n} \left(C_i^{\tilde{\Delta}} \, \mathrm{e}^{-D\lambda_i t} + \int_0^t \mathrm{e}^{-D\lambda_i (t-\tau)} \langle \mathcal{D}^{1/2} \, \Gamma(\tau,\cdot), V_i \rangle_G \, \mathrm{d}\tau \right) V_i(x) \tag{34}$$

with $(C_1^{\tilde{\Delta}}, C_2^{\tilde{\Delta}}, \dots, C_n^{\tilde{\Delta}}) \in \mathbb{R}^n$.

It is worth pointing out that these analytical solutions easily are implementable in a computer.

Proof. We first focus on the solution for Equation (21). We decompose $f(t,x)$ into the basis $(v_i)_{i \in \{1,2,\dots,n\}}$ of $\mathbb{R}^n$ made of eigenvectors of the matrix L.

$$f(t,x) = \sum_{i=1}^{n} \alpha_i(t) \, v_i(x), \tag{35}$$

where, for all $i \in \{1,2,\dots,n\}$, we have

$$\alpha_i(t) = \langle f, v_i \rangle_G.$$

We have

$$\langle f, L v_i \rangle_G = \mu_i \langle f, v_i \rangle_G = \mu_i \, \alpha_i(t).$$

Using Lemma 1, we have

$$\mu_i \, \alpha_i(t) = \langle L f, v_i \rangle_G = \langle L f, v_i \rangle_G,$$

and using Equation (21), we have

$$-D \, \mu_i \, \alpha_i(t) = \left\langle \frac{\partial f}{\partial t}(t,\cdot), v_i \right\rangle_G - \langle \Gamma(t,\cdot), v_i \rangle_G.$$

Implementing Equation (35), we have

$$\begin{aligned}
-D \, \mu_i \, \alpha_i(t) &= \frac{\mathrm{d}}{\mathrm{d}t} \sum_{j=1}^{n} \alpha_j(t) \langle v_j, v_i \rangle_G - \langle \Gamma(t,\cdot), v_i \rangle_G. \\
\iff D \, \mu_i \, \alpha_i(t) &= -\frac{\mathrm{d}}{\mathrm{d}t} \sum_{j=1}^{n} \alpha_j(t) \, \delta_{ij} + \langle \Gamma(t,\cdot), v_i \rangle_G \\
&= -\frac{\mathrm{d}\alpha_i(t)}{\mathrm{d}t} + \langle \Gamma(t,\cdot), v_i \rangle_G. \\
\iff \frac{\mathrm{d}\alpha_i(t)}{\mathrm{d}t} &= -D \, \mu_i \, \alpha_i(t) + \langle \Gamma(t,\cdot), v_i \rangle_G.
\end{aligned}$$

We can easily solve this differential equation, and we have

$$\alpha_i(t) = \alpha_i(0) \, \mathrm{e}^{-D\mu_i t} + \int_0^t \mathrm{e}^{-D\mu_i (t-\tau)} \langle \Gamma(\tau,\cdot), v_i \rangle_G \, \mathrm{d}\tau.$$

We obtain the final result by placing the previous equation to Equation (35).

We now focus on the solution for Equation (22). We decompose $(\mathcal{D}^{1/2}f)(t,x)$ into the basis $(V_i)_{i\in\{1,2,\ldots,n\}}$ of $\mathbb{R}^n$ made of eigenvectors of the matrix $\tilde{\Delta}$. The rest of the calculation remains exactly the same as below, with the additional trick $\tilde{\Delta}\mathcal{D}^{1/2} = \mathcal{D}^{1/2}\Delta$. □

We note that $C_i^L = C_i^{\tilde{\Delta}}/\sqrt{\mathrm{d}_A x}$, for any $i \in [\![1,n]\!]$ and $x \in G$. It turns out that if $\Gamma = 0$, using L or Δ leads to the exact same result.

Corollary 1. *The solution for the diffusion Equation (21) is written as*

$$f(t,x) = \langle f(0,\cdot), \mathcal{U}^L(t,x,\cdot)\rangle_G + \int_0^t \langle \Gamma(t-\tau,\cdot), \mathcal{U}^L(\tau,x,\cdot)\rangle_G \,\mathrm{d}\tau \tag{36}$$

$$f(t,x) = \langle f(0,\cdot), \mathcal{U}^{\tilde{\Delta}}(t,x,\cdot)\rangle_G + \int_0^t \langle \Gamma(t-\tau,\cdot), \mathcal{U}^{\tilde{\Delta}}(\tau,x,\cdot)\rangle_G \,\mathrm{d}\tau \tag{37}$$

where $\mathcal{U}^L : \mathbb{R}_+ \times E^2 \to \mathbb{R}$ *is the diffusion propagator for the matrix L, and is such that*

$$\mathcal{U}^L(t,x,y) = \sum_{i=1}^n \mathrm{e}^{-D\mu_i t}\, v_i(x)\, v_i(y), \quad \forall (t,x,y) \in \mathbb{R}_+ \times E^2. \tag{38}$$

We also have

$$\mathcal{U}^{\tilde{\Delta}}(t,x,y) = \sqrt{\frac{\mathrm{d}_A y}{\mathrm{d}_A x}} \sum_{i=1}^n \mathrm{e}^{-D\lambda_i t}\, V_i(x)\, V_i(y), \quad \forall (t,x,y) \in \mathbb{R}_+ \times E^2. \tag{39}$$

The advantage of this last expression is the explicit dependency of f with respect to the initial condition on all the nodes of G, i.e., with $f(0,y)$ for all $y \in G$.

Proof. We derive the expression for the matrix $\tilde{\Delta}$. From the general solution expression, we have

$$f(0,x) = \frac{1}{\sqrt{\mathrm{d}_A x}} \sum_{j=1}^n C_j\, V_j(x).$$

Therefore, we have

$$C_i = \langle V_i, \sqrt{\mathrm{d}_A} f(0,\cdot)\rangle_G = \sum_{y\in G} V_i(y)\, \sqrt{\mathrm{d}_A y}\, f(0,y).$$

Implementing into the main solution expression, we have

$$\begin{aligned} f(t,x) =& \frac{1}{\sqrt{\mathrm{d}_A x}} \sum_{i=1}^n \left(\sum_{y\in G} V_i(y)\, \sqrt{\mathrm{d}_A y}\, f(0,y)\, \mathrm{e}^{-D\lambda_i t} \right. \\ &\left. + \int_0^t \sum_{y\in G} \sqrt{\mathrm{d}_A y} V_i(y) \Gamma(\tau,y)\, \mathrm{e}^{-D\lambda_i (t-\tau)}\,\mathrm{d}\tau \right) V_i(x). \end{aligned}$$

Some re-ordering leads to

$$\begin{aligned} f(t,x) =& \sum_{y\in G} \sqrt{\frac{\mathrm{d}_A y}{\mathrm{d}_A x}} \sum_{i=1}^n V_i(x) V_i(y)\, f(0,y)\, \mathrm{e}^{-D\lambda_i t} \\ &+ \int_0^t \sum_{y\in G} \sqrt{\frac{\mathrm{d}_A y}{\mathrm{d}_A x}} \sum_{i=1}^n V_i(x) V_i(y) \Gamma(\tau,y)\, \mathrm{e}^{-D\lambda_i (t-\tau)}\,\mathrm{d}\tau. \end{aligned}$$

This explicitly is what we are looking for. A variable change will lead to the result. □

Remark 1. *We have*

$$\mathcal{U}^{\tilde{\Delta}}(0,x,y) = \sqrt{\frac{\mathrm{d}_A y}{\mathrm{d}_A x}} \sum_{i=1}^{n} V_i(x)\, V_i(y) = \sqrt{\frac{\mathrm{d}_A y}{\mathrm{d}_A x}}\, \delta_{xy} = \delta_{xy},$$

where $\delta_{xy} = 1$ if $x = y$ and 0 otherwise: the same for matrix L. We also have

$$\lim_{t\to+\infty} \mathcal{U}^{\tilde{\Delta}}(t,x,y) = \sqrt{\frac{\mathrm{d}_A y}{\mathrm{d}_A x}}\, V_n(x)\, V_n(y) = \sqrt{\frac{\mathrm{d}_A y}{\mathrm{d}_A x}} \sqrt{\frac{\mathrm{d}_A x}{\mathrm{Vol}(G)}} \sqrt{\frac{\mathrm{d}_A y}{\mathrm{Vol}(G)}} = \frac{\mathrm{d}_A y}{\mathrm{Vol}(G)},$$

in which case $\lim_{t\to+\infty} \mathcal{U}^{\tilde{\Delta}}(t,x,y)$ does not depend on x. We also found an interesting interpretation for the eigenvector V_n. Indeed, we have

$$\lim_{t\to+\infty} \mathcal{U}^{\tilde{\Delta}}(t,x,y) = V_n^2(y).$$

Therefore, $V_n^2(y)$ represents the propagation of information due to the diffusion, to node y.

The next two sections deal with further developments on the implications of the facts established above. This also holds for the operator L.

4. Standard Equilibrium and Blockchain Equilibrium

The equilibrium is an important aspect of the current paper. We identified two types of equilibrium, which we further develop here. The zero eigenvalue as well as its eigenvector will play an important role, as we are going to see.

4.1. Standard Equilibrium

4.1.1. Integrable Creation Destruction Function

The standard equilibrium corresponds to the steady state, i.e., when $t \to +\infty$. It is expected to have permanent exchanges between nodes at the same rate, which is exactly equivalent to the fact that each node keeps having the same quantity of information, i.e., the same quantity of hash generation.

More specifically, we have the following proposition.

Theorem 3 (Standard Equilibrium Framework)**.** *Let f be solution of the diffusion Equation (1). We assume that $t \mapsto \Gamma(t,y)$ is integrable on $\mathbb{R}_+$ for any $y \in G$, and*

$$\int_0^t \Gamma(\tau,y)\, \mathrm{e}^{-D\lambda_i(t-\tau)}\, \mathrm{d}\tau \underset{(t\to+\infty)}{\sim} \gamma_i(y), \quad \text{with } 0 \le \gamma_i(y) < +\infty, \quad \forall y \in G. \tag{40}$$

Then the standard equilibrium state corresponds to:

$$\lim_{t\to+\infty} f(t,x) = \frac{1}{\mathrm{Vol}(G)} \sum_{y\in G} \mathrm{d}_A y \left(f(0,y) + \int_0^{+\infty} \Gamma(\tau,y)\, \mathrm{d}\tau \right). \tag{41}$$

We have a similar result by considering the matrix L.

Proof. From the expression

$$\mathcal{U}(t,x,y) = \sqrt{\frac{\mathrm{d}_A y}{\mathrm{d}_A x}} \sum_{i=1}^{n} \mathrm{e}^{-D\lambda_i t}\, V_i(x)\, V_i(y),$$

we see that all the terms cancel since $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_{n-1} > \lambda_n = 0$, *except* the last one. Thus

$$\lim_{t \to +\infty} \mathcal{U}(t, x, y) = \sqrt{\frac{\mathrm{d}_A y}{\mathrm{d}_A x}}\, V_n(x)\, V_n(y) = \frac{\mathrm{d}_A y}{\mathrm{Vol}(G)}.$$

This does not depend on x. We then start from the following equation

$$\begin{aligned} f(t,x) = & \sum_{y \in G} \sqrt{\frac{\mathrm{d}_A y}{\mathrm{d}_A x}} \sum_{i=1}^{n} V_i(x) V_i(y)\, f(0,y)\, \mathrm{e}^{-D\lambda_i t} \\ & + \int_0^t \sum_{y \in G} \sqrt{\frac{\mathrm{d}_A y}{\mathrm{d}_A x}} \sum_{i=1}^{n} V_i(x) V_i(y) \Gamma(\tau, y)\, \mathrm{e}^{-D\lambda_i (t-\tau)}\, \mathrm{d}\tau. \end{aligned}$$

Hence,

$$\begin{aligned} \lim_{t \to +\infty} f(t,x) = & \frac{1}{\mathrm{Vol}(G)} \sum_{y \in G} \mathrm{d}_A y\, f(0,y) \\ & + \sum_{y \in G} \sqrt{\frac{\mathrm{d}_A y}{\mathrm{d}_A x}} \left[\sum_{i=1}^{n} V_i(x) V_i(y) \left(\lim_{t \to +\infty} \int_0^t \Gamma(\tau, y)\, \mathrm{e}^{-D\lambda_i (t-\tau)}\, \mathrm{d}\tau \right) \right]. \end{aligned}$$

The previous equation is the most general one. We come to the factor between parenthesis. The previous equation is then

$$\lim_{t \to +\infty} f(t,x) = \frac{1}{\mathrm{Vol}(G)} \sum_{y \in G} \mathrm{d}_A y\, f(0,y) + \sum_{y \in G} \sqrt{\frac{\mathrm{d}_A y}{\mathrm{d}_A x}} \left[\sum_{i=1}^{n} V_i(x) V_i(y)\, \gamma_i(y) \right].$$

We now focus on the Equations (40). We will prove that $\gamma_i(y) = 0$, for all $i \in \{1, \ldots, n-1\}$ and $y \in G$, which will prove Equation (41). A variable change gives

$$\int_0^t \Gamma(\tau, y)\, \mathrm{e}^{-D\lambda_i (t-\tau)}\, \mathrm{d}\tau = \int_0^t \Gamma(t-\tau, y)\, \mathrm{e}^{-D\lambda_i \tau}\, \mathrm{d}\tau.$$

Bearing this in mind, for $i = n$, we have

$$\int_0^t \Gamma(\tau, y)\, \mathrm{d}\tau \underset{(t \to +\infty)}{\sim} \gamma_n(y),$$

that is

$$\int_0^{+\infty} \Gamma(\tau, y)\, \mathrm{d}\tau = \gamma_n(y) < +\infty.$$

Hence $|\Gamma(\tau, y)| \leq C(y) < +\infty$ with C function from G to $\mathbb{R}$, and $\lim\limits_{\tau \to +\infty} \Gamma(\tau, y) = 0$, for any $y \in G$. It turns out that for any $i \in \{1, 2, \ldots, n-1\}$, we have

$$\forall \epsilon > 0 \quad \exists \tilde{\tau} > 0 \quad \forall \tau > \tilde{\tau} \quad |\Gamma(\tau, y)| < \epsilon \frac{D\lambda_i}{C(y) + 1}, \quad \forall y \in G.$$

Hence, for any $i \in \{1, 2, \ldots, n-1\}$ and $t > \tau$, we have

$$\left| \int_{\tilde{\tau}}^t \Gamma(\tau, y)\, \mathrm{e}^{-D\lambda_i (t-\tau)}\, \mathrm{d}\tau \right| < \int_{\tilde{\tau}}^t |\Gamma(\tau, y)|\, \mathrm{e}^{-D\lambda_i (t-\tau)}\, \mathrm{d}\tau < \epsilon \frac{1 - \mathrm{e}^{-D\lambda_i (t-\tilde{\tau})}}{C(y) + 1}.$$

Hence

$$\left| \int_0^t \Gamma(\tau, y)\, \mathrm{e}^{-D\lambda_i (t-\tau)}\, \mathrm{d}\tau - \int_0^{\tilde{\tau}} \Gamma(\tau, y)\, \mathrm{e}^{-D\lambda_i (t-\tau)}\, \mathrm{d}\tau \right| < \epsilon \frac{1 - \mathrm{e}^{-D\lambda_i (t-\tilde{\tau})}}{C(y) + 1}.$$

By a variable change and using triangular inequality, we have

$$\left|\int_0^t \Gamma(\tau,y)\,\mathrm{e}^{-D\lambda_i(t-\tau)}\,\mathrm{d}\tau\right| < \left|\int_{t-\tilde{\tau}}^t \Gamma(t-\tau,y)\,\mathrm{e}^{-D\lambda_i\tau}\,\mathrm{d}\tau\right| + \epsilon\,\frac{1-\mathrm{e}^{-D\lambda_i(t-\tilde{\tau})}}{C(y)+1}.$$

In other words, we have

$$\left|\int_0^t \Gamma(\tau,y)\,\mathrm{e}^{-D\lambda_i(t-\tau)}\,\mathrm{d}\tau\right| < C(y)\,\frac{\mathrm{e}^{-D\lambda_i t}-\mathrm{e}^{-D\lambda_i(t-\tilde{\tau})}}{D\lambda_i} + \epsilon\,\frac{1-\mathrm{e}^{-D\lambda_i(t-\tilde{\tau})}}{C(y)+1}.$$

Suppose now that $t \in \mathbb{R}_+^*$ is such that

$$t = \max\left(-\frac{1}{D\lambda_i}\ln\left(\epsilon\,\frac{D\lambda_i}{C(y)+1}\right),\,\tau\right).$$

Then we have

$$\left|\int_0^t \Gamma(\tau,y)\,\mathrm{e}^{-D\lambda_i(t-\tau)}\,\mathrm{d}\tau\right| < \epsilon.$$

Therefore,

$$\lim_{t\to+\infty}\int_0^t \Gamma(\tau,y)\,\mathrm{e}^{-D\lambda_i(t-\tau)}\,\mathrm{d}\tau = \gamma_i(y) = 0,$$

which concludes the proof. □

For example, set $\Gamma(t,x) = P(t)\,\mathrm{e}^{\alpha t}$, where P is a polynomial and $\alpha \in \mathbb{R}$ with some restrictions that we need to discover.

Let $n \in \mathbb{N}$. A quick calculation allows to establish that

$$\int \tau^n\,\mathrm{e}^{a\tau}\,\mathrm{d}\tau = \begin{cases} \displaystyle\sum_{j=0}^{n}(-1)^{n-j}\left(\frac{1}{a}\right)^{n-j+1}\frac{n!}{j!}\,\tau^{n-j}\,\mathrm{e}^{a\tau} + \mathrm{C}^{\mathrm{te}} & \text{if } a \neq 0 \\ \displaystyle\frac{\tau^{n+1}}{n+1} + \mathrm{C}^{\mathrm{te}} & \text{otherwise} \end{cases}$$

If we set $\Gamma(\tau,x) = \tau^n\,\mathrm{e}^{\alpha\tau}$, then we deduce that

$$\mathrm{e}^{-D\lambda_i t}\int_0^t \tau^n\,\mathrm{e}^{(D\lambda_i+\alpha)\tau}\,\mathrm{d}\tau = \begin{cases} \displaystyle\sum_{j=0}^{n}(-1)^{n-j}\left(\frac{1}{D\lambda_i+\alpha}\right)^{n-j+1}\frac{n!}{j!}\,t^{n-j}\,\mathrm{e}^{\alpha t} & \text{if } \alpha \neq -D\lambda_i \\ \displaystyle\mathrm{e}^{-D\lambda_i t}\,\frac{t^{n+1}}{n+1} & \text{otherwise} \end{cases}$$

We deduce that, for $\gamma_i(y)$ to be finite, we need to have $\alpha \in \mathbb{R}_-^*$, while $n \in \mathbb{N}$ without restriction, and the degree of P is in $\mathbb{N}$. However, if $\alpha = 0$, then we necessarily need to have $n = 0$. The less negligible function Γ is thus $\Gamma(t,x) = P(0)$, which is the only form giving a nonzero $\gamma_i(y) = 0$.

Remark 2. *Equation (40) is actually close to the one found by applying the mean value theorem, provided $\Gamma(\cdot,y)$ is continuous for any $y \in G$: there exists $a_i \in [0,t]$ such that*

$$\int_0^t \Gamma(\tau,y)\,\mathrm{e}^{D\lambda_i\tau}\,\mathrm{d}\tau = \frac{\Gamma(a_i,y)}{D\lambda_i}\,(\mathrm{e}^{D\lambda_i t}-1), \quad \forall y \in G. \tag{42}$$

Thus, $\gamma_i(y) = \Gamma(a_i,y)/D\lambda_i$, with $a \in \mathbb{R}_+^*$.

Remark 3. *General function theory might be interesting if one decides to break the assumption that $\Gamma(\cdot,y)$ is continuous.*

4.1.2. Continuous-by-Parts Creation Destruction Function

If we require the solution f to be continuous at all times, then, according to the diffusion equation, Γ must be at least *continuous by parts*. In this case, the function $\Gamma(\cdot, x)$ is continuous by parts, for any $x \in E$, i.e., there are some points on the time axis $\mathbb{R}_+$ where the function $\Gamma(\cdot, x)$ is not continuous (discontinuity of first kind). In this case, for any $x \in G$, the function $\Gamma(\cdot, x)$ still is integrable on $\mathbb{R}_+$.

4.2. Blockchain Equilibrium (Approximation)

In fact, we have to consider that time is *discretized*, i.e., subdivided into interval of time which are of length 10 min approximately. Ten minutes is the average time needed for a miner to mine a new block in the Bitoin blockchain Lipton and Treccani (2021), and this could be changed to 15 s for the Ethereum blockchain. We will go to deeper detail to this point in the future, but approximately every 10 min, the *blockchain equilibrium equation* is satisfied. The following are functions of interest:

- r_b—b-block reward;
- c—cost per hash, supposed to be constant;
- P_t—price of one Bitcoin, in USD, at time t;
- f_i—fee for transaction $i \in I$, where I is the set of all transactions that will be stacked into block b;
- N_b—number of hashes in the network, for block b.

From these five variables, we can easily compute the revenues R_b for block b, as follows.

$$R_b = \left(r_b + \sum_{i \in I} f_i \right) P_T. \tag{43}$$

We can compose the costs, more precisely the expected costs C_b, for block b, as well:

$$C_b = c\, N_b. \tag{44}$$

We have the *blockchain equilibrium* equation:

$$R_b = C_b \Leftrightarrow \left(r_b + \sum_{i \in I} f_i \right) P_T = c\, N_b. \tag{45}$$

Thus, N_b is connected to $f(t, x)$, which gives the cost for forming a block. We can, in fact, make the assumption that the circulation of transaction lists within miners is very fast: it takes approximately 1 millisecond for the nodes to receive the transaction list from an original list. It is relevant to assume that the diffusion happens very fast once one miner has a transaction list to be sent, and that the diffusion coefficient D is very high. This is the *blockchain equilibrium* approximation: $D\lambda_{n-1}t \gg 1$, for any time t.

In other words, $D\lambda_{n-1}t$ is so high that $\mathrm{e}^{-D\lambda_i t} \underset{(D\lambda_{n-1}t \to +\infty)}{\sim} 0$, for any $i \in \{1, 2, \ldots, n-1\}$. Only the term corresponding to $\lambda_n = 0$ survives.

Proposition 6 (Blockchain Equilibirum Approximation)**.** *Let f be solution of the diffusion Equation (1). We focus on $t \in [0, T]$, where T is set to be 10 min. Then the blockchain equilibrium state corresponds to*

$$f(T, x) \underset{(D\lambda_{n-1}t \to +\infty)}{\sim} \frac{1}{\mathrm{Vol}(G)} \sum_{y \in G} \mathrm{d}_A y \left(f(0, y) + \int_0^T \Gamma(\tau, y)\, \mathrm{d}\tau \right). \tag{46}$$

We have a similar result by considering the matrix L.

It turns out that $f(T, x)$, which does not depend on x, is the same for all the nodes.

Proof. We have

$$\mathrm{e}^{-D\lambda_i \tau} \underset{(D\lambda_{n-1}t\to+\infty)}{\sim} \begin{cases} 0 & \text{if } i \in \{1,2,\dots,n-1\} \\ 1 & \text{if } i = n \end{cases}$$

and

$$\int_0^t \Gamma(\tau,y)\,\mathrm{e}^{-D\lambda_i(t-\tau)}\,\mathrm{d}\tau \underset{(D\lambda_{n-1}t\to+\infty)}{\sim} \begin{cases} 0 & \text{if } i \in \{1,2,\dots,n-1\} \\ \displaystyle\int_0^{+\infty} \Gamma(\tau,y)\,\mathrm{d}\tau & \text{if } i = n \end{cases}$$

We deduce the equation from

$$\begin{aligned} f(t,x) = {} & \sum_{y\in G} \sqrt{\frac{\mathrm{d}_A y}{\mathrm{d}_A x}} \sum_{i=1}^n V_i(x)V_i(y)\, f(0,y)\,\mathrm{e}^{-D\lambda_i t} \\ & + \sum_{y\in G} \sqrt{\frac{\mathrm{d}_A y}{\mathrm{d}_A x}} \left[\sum_{i=1}^n V_i(x)V_i(y) \left(\int_0^t \Gamma(\tau,y)\,\mathrm{e}^{-D\lambda_i(t-\tau)}\,\mathrm{d}\tau \right) \right], \end{aligned}$$

which concludes the proof. □

Thus, $f(T,x)$ is to be associated to N_b. Finally, in essence, the standard equilibrium and blockchain equilibrium give the same results. In the following, '$f(T,\cdot)$' refers to the function at the equilibrium state.

5. Undirected Graph: Case Studies

5.1. 2 Nodes

As a quick application, suppose $E = \{x,y\}$, only two nodes, and $V = \{(x,y)\}$, one connection. Suppose $\Gamma = 0$, $f(0,x) = f_0 \in \mathbb{R}_+$, $f(0,y) = 0$. We have $V_1(x)^2 + V_1(y)^2 = 1$, $\lambda_2 = 0$, and

$$\begin{aligned} f(t,x) &= f_0\,\mathcal{U}(t,x,x) = f_0\,V_1(x)^2\,\mathrm{e}^{-D\lambda_1 t} + f_0\,V_2(x)^2, \\ f(t,y) &= f_0\,\mathcal{U}(t,y,x) = f_0\,V_1(x)\,V_1(y)\,\mathrm{e}^{-D\lambda_1 t} + f_0\,V_2(x)\,V_2(y). \end{aligned}$$

We can calculate the eigenvalues and eigenvectors for the network. The non-normalized Laplacian matrix is given by

$$L = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$$

and the Laplacian matrix by $\Delta = \tilde{\Delta} = L$. The eigenvalues are $\lambda = 2$ and $\lambda_2 = 0$, and the eigenvectors are $V_1 = {}^t(1/\sqrt{2} \quad -1/\sqrt{2})$ and $V_2 = {}^t(1/\sqrt{2} \quad 1/\sqrt{2})$. We finally obtain

$$\begin{cases} f(t,x) & = \dfrac{f_0}{2}\left(1 + \mathrm{e}^{-2Dt}\right) \\ \\ f(t,y) & = \dfrac{f_0}{2}\left(1 - \mathrm{e}^{-2Dt}\right) \end{cases}$$

See Figure 3 for a plot of these two solutions.

The equilibrium is therefore given by

$$f(T,x) = f(T,y) = f_0/2.$$

Finally, due to the diffusion effect, both nodes share the same information at the equilibrium.

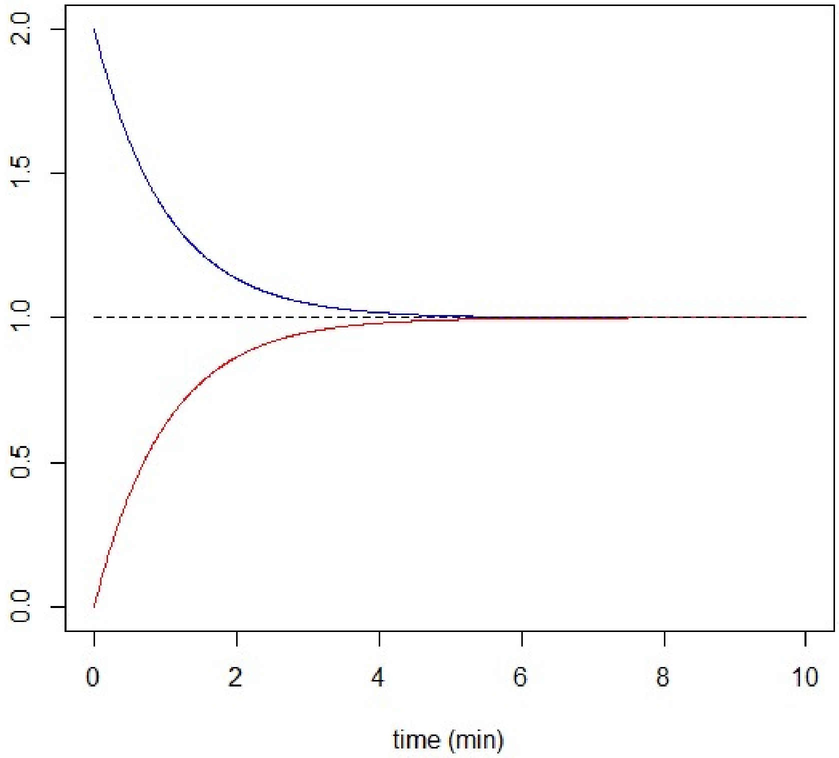

Figure 3. Illustration of the diffusion information for 2 nodes. The initial conditions are $\Gamma = 0$, $f(0,x) = f_0 \in \mathbb{R}_+$, and $f(0,y) = 0$. We set $D = 1/2$, and $f_0 = 2$. Blue is for $f(t,x)$, while red is for $f(t,y)$.

5.2. 3 Nodes

We set that $E = \{1,2,3\}$ and $V = \{(1,2),(2,3)\}$. The non-normalized Laplacian of such a matrix is

$$L = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{pmatrix}$$

and is

$$\tilde{\Delta} = \mathcal{D}^{-1/2}\, L\, \mathcal{D}^{-1/2} = \begin{pmatrix} 1 & -1/\sqrt{2} & 0 \\ -1/\sqrt{2} & 2 & -1/\sqrt{2} \\ 0 & -1/\sqrt{2} & 1 \end{pmatrix}.$$

The eigenvalues of $\tilde{\Delta}$ are the roots of the polynomial $(1-X)\left[(1-X)^2 - (1-X)/2 - 1/2\right]$, that is $\lambda_1 = 2$, $\lambda_2 = 1$, and $\lambda_3 = 0$. The eigenvectors are $V_1 = {}^t(1/2 \quad -1/\sqrt{2} \quad 1/2)$, $V_2 = {}^t(1/\sqrt{2} \quad 0 \quad -1/\sqrt{2})$, and $V_3 = {}^t(1/2 \quad 1/\sqrt{2} \quad 1/2)$.

We set $\Gamma = 0$. In order to be very explicit, the propagators are written as, for $x = 1$,

$$\begin{cases} \mathcal{U}(t,1,1) &= \dfrac{1}{4} + \dfrac{1}{2}\mathrm{e}^{-Dt} + \dfrac{1}{4}\mathrm{e}^{-2Dt} \\[2ex] \mathcal{U}(t,1,2) &= \dfrac{1}{2} - \dfrac{1}{2}\mathrm{e}^{-2Dt} \\[2ex] \mathcal{U}(t,1,3) &= \dfrac{1}{4} - \dfrac{1}{2}\mathrm{e}^{-Dt} + \dfrac{1}{4}\mathrm{e}^{-2Dt} \end{cases}$$

for $x = 2$

$$\begin{cases} \mathcal{U}(t,2,1) &= \frac{1}{4} - \frac{1}{4}\mathrm{e}^{-2Dt} \\ \mathcal{U}(t,2,2) &= \frac{1}{2} + \frac{1}{2}\mathrm{e}^{-2Dt} \\ \mathcal{U}(t,2,3) &= \frac{1}{4} - \frac{1}{4}\mathrm{e}^{-2Dt} \end{cases}$$

and for $x = 3$

$$\begin{cases} \mathcal{U}(t,1,1) &= \frac{1}{4} - \frac{1}{2}\mathrm{e}^{-Dt} + \frac{1}{4}\mathrm{e}^{-2Dt} \\ \mathcal{U}(t,1,2) &= \frac{1}{2} - \frac{1}{2}\mathrm{e}^{-2Dt} \\ \mathcal{U}(t,1,3) &= \frac{1}{4} + \frac{1}{2}\mathrm{e}^{-Dt} + \frac{1}{4}\mathrm{e}^{-2Dt} \end{cases}$$

Therefore, we have

$$\begin{cases} f(t,1) &= f(0,1)\left(\frac{1}{4} + \frac{1}{4}\mathrm{e}^{-Dt} + \frac{1}{4}\mathrm{e}^{-2Dt}\right) \\ &\qquad +f(0,2)\left(\frac{1}{2} - \frac{1}{2}\mathrm{e}^{-2Dt}\right) \\ &\qquad\qquad +f(0,3)\left(\frac{1}{4} - \frac{1}{4}\mathrm{e}^{-Dt} + \frac{1}{4}\mathrm{e}^{-2Dt}\right) \\ f(t,2) &= f(0,1)\left(\frac{1}{4} - \frac{1}{4}\mathrm{e}^{-2Dt}\right) \\ &\qquad +f(0,2)\left(\frac{1}{2} + \frac{1}{2}\mathrm{e}^{-2Dt}\right) \\ &\qquad\qquad +f(0,3)\left(\frac{1}{4} - \frac{1}{4}\mathrm{e}^{-2Dt}\right) \\ f(t,3) &= f(0,1)\left(\frac{1}{4} - \frac{1}{4}\mathrm{e}^{-Dt} + \frac{1}{4}\mathrm{e}^{-2Dt}\right) \\ &\qquad +f(0,2)\left(\frac{1}{2} - \frac{1}{2}\mathrm{e}^{-2Dt}\right) \\ &\qquad\qquad +f(0,3)\left(\frac{1}{4} + \frac{1}{4}\mathrm{e}^{-Dt} + \frac{1}{4}\mathrm{e}^{-2Dt}\right) \end{cases}$$

Figure 4 plots the evolution of the three functions.

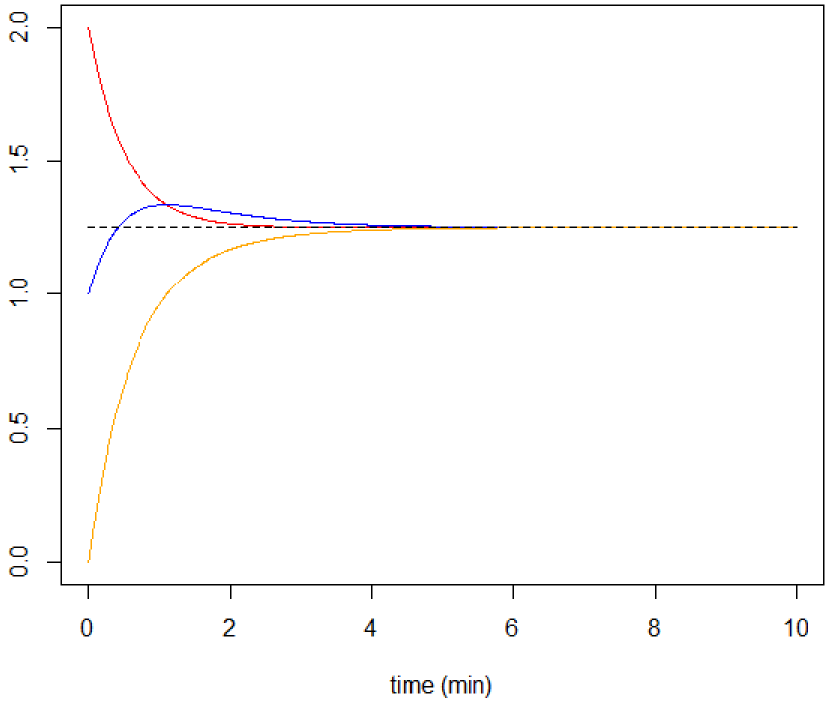

Figure 4. Illustration of the diffusion information for 3 nodes. Node 2 is connected to nodes 1 and 3, but nodes 1 and 3 are not connected. The initial conditions are $\Gamma = 0$, $f(0,1) = 1$, $f(0,2) = 2$ and $f(0,3) = 0$. We set $D = 1$. Blue is for $f(t,1)$, red is for $f(t,2)$, and yellow is for $f(t,3)$.

The equilibrium state corresponds to the following expressions.

$$f(T,1) = f(T,2) = f(T,3) = \frac{1}{4}f(0,1) + \frac{1}{2}f(0,2) + \frac{1}{4}f(0,3).$$

We are now going to propose the solutions for specific P2P networks. It is worth stressing that implementation of the above in a prototype is in progress.

6. Generalization to Directed Graphs

This section focuses on the generalization of what has been seen so far, but applied to a directed graph. Concretely, information between different layers of the whole Bitcoin network is directed. A very simple example is that the users layer is sending information to the P2P network as transaction lists, and they do not expect any information back to them since they are not miners. The diffusion is thus directed. This breaks the symmetry of the adjacency matrix, which is not diagonalizable anymore. Thus, we could handle a more general framework to perform the diffusion. So far, we may handle with straight numerical solutions for the implementation, but we are going to see that we can derive an analytical solution base on the singular value decomposition (SVD).

Definition 8 (SVD)**.** *Let $A \in \mathcal{M}_{n,m}(\mathbb{R})$ be a matrix of size $n \times m$, with $n, m \in \mathbb{N}^*$. Then, A admits the following decomposition:*

$$A = U\Sigma\,^tV, \tag{47}$$

where U and V are orthogonal square matrices of size n and m, respectively, and $\Sigma \in \mathcal{M}_{n,m}(\mathbb{R})$ is a 0 matrix, except its diagonal elements, which are non-negative numbers.

If $m > n$, we have

$$\Sigma = \begin{pmatrix} \sigma_1 & 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \sigma_n & 0 & \dots & 0 \end{pmatrix}.$$

If Σ is a diagonal matrix $\Sigma = \mathrm{diag}(\sigma_1, \dots, \sigma_n)$. The numbers $\sigma_1 \geq \dots \geq \sigma_n \geq 0$ are the *singular values* of A. The columns of matrices U and V are the *singular vectors* of A.

6.1. Matrix Exponential and Adjoint of the Laplacian Operator

As a reminder, we need the following three definitions.

Definition 9 (Exponential of Matrix). *Let $A \in \mathcal{M}_n(\mathbb{R})$ for $n \in \mathbb{N}^*$. The* exponential *of the matrix A is given by*

$$\exp(A) = \sum_{k=0}^{+\infty} \frac{A^k}{k!}. \tag{48}$$

We write e^A for $\exp(A)$.

Definition 10 (Hyperbolic Cosine and Sine of Matrix). *Let $A \in \mathcal{M}_n(\mathbb{R})$ for $n \in \mathbb{N}^*$. The hyperbolic cosine of the matrix A is given by*

$$\mathrm{ch}(A) = \frac{1}{2}\,(\mathrm{e}^A + \mathrm{e}^{-A}). \tag{49}$$

The hyperbolic sine of the matrix M is given by

$$\mathrm{sh}(A) = \frac{1}{2}\,(\mathrm{e}^A - \mathrm{e}^{-A}). \tag{50}$$

It turns out that

$$\mathrm{ch}(A) = \sum_{k=0}^{+\infty} \frac{A^{2k}}{(2k)!},$$

$$\mathrm{sh}(A) = \sum_{k=0}^{+\infty} \frac{A^{2k+1}}{(2k+1)!}.$$

Definition 11. *Let T be a (linear) operator and $g \subseteq G$ be a subgraph. The* adjoint *operator ${}^tT$ of T is such that*

$$\langle h, Tf \rangle_g = \langle {}^tTh, f \rangle_g, \tag{51}$$

for any $f, g : G \to \mathbb{R}$. The adjoint operator is the transposed of the associated matrix.

6.2. Solution for the Directed Diffusion Involving the Operator L

6.2.1. Intuition

In case of the directed network, the diffusion also is oriented. To gain intuition, suppose there are two nodes x and y, so that x is connected to y, but not the converse. In this case, the adjacency matrix writes

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$$

so that the non-normalized Laplacian matrix is

$$L = \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix}.$$

In addition, the node x loses information, while y gains the same amount of information that x loses. Therefore, at any time $t \in \mathbb{R}_+$, the conservation of total information is satisfied:

$$f(t,x) + f(t,y) = f(0,x) + f(0,y) = \mathrm{C}^{\mathrm{te}}.$$

Since x loses information but in a diffusive effect such that the more information it has, the more it gives, we must have

$$\frac{\partial f}{\partial t}(t,x) = -D\, f(t,x).$$

Solving this equation leads to

$$f(t,x) = f(0,x)\, \mathrm{e}^{-D\,t},$$

so that

$$\mathrm{C}^{\mathrm{te}} - f(t,y) = (\mathrm{C}^{\mathrm{te}} - f(0,y))\, \mathrm{e}^{-D\,t} \Longrightarrow f(t,y) = \mathrm{C}^{\mathrm{te}}(1 - \mathrm{e}^{-D\,t}) + f(0,y)\, \mathrm{e}^{-D\,t}.$$

This is the solution of the differential equation:

$$\frac{\partial f}{\partial t}(t,y) = D\, f(t,x).$$

Introducing the vector $F(t) = {}^{t}(f(t,x)\ f(t,y))$, we thus obtain the following system of differential equations:

$$\frac{\mathrm{d}F}{\mathrm{d}t}(t) = -D\, {}^{t}L\, F(t).$$

6.2.2. The Directed Diffusion Equation Involving L

Definition 12. *Let $f : E \times \mathbb{R}_+ \to \mathbb{R}$ representing the information function for a graph with n nodes. If D is the diffusion coefficient and $\Gamma : E \times \mathbb{R}_+ \to \mathbb{R}$ the creation function of the information, the following directed diffusion equation holds, for any $t \in \mathbb{R}_+$:*

$$\frac{\partial f}{\partial t}(t,x) = -D\, {}^{t}L\, f(t,x) + \Gamma(t,x) \tag{52}$$

We have the same Equation (21) if and only if $L = {}^{t}L$, that is, if and only if A is symmetrical.

6.2.3. Solution Involving L

From Equation (12), we can use the Cauchy theorem to obtain the following solution.

Theorem 4 (Application of the Cauchy Theorem)**.** *The solution of the related Cauchy problem associated with Equation (12) is given by*

$$f(t,x) = \mathrm{e}^{-D\, {}^{t}L\, t}\, f(0,x) + \int_0^t \mathrm{e}^{-D\, {}^{t}L\,(t-\tau)}\, \Gamma(\tau,x)\, \mathrm{d}\tau. \tag{53}$$

This expression is straightforwardly implementable in a computer. As a tip, we would rather use the following equation:

$$f(t+\Delta t,x) = \mathrm{e}^{-D\, {}^{t}L\, \Delta t}\, f(t,x) + \int_t^{t+\Delta t} \mathrm{e}^{-D\, {}^{t}L\,(t-\tau)}\, \Gamma(\tau,x)\, \mathrm{d}\tau. \tag{54}$$

The advantage of this equation is to force step-by-step dependency of configuration at time $t - \Delta t$, for any time t, for the increment $\Delta t > 0$. See Section 8.3 for a practical implementation from this equation.

6.2.4. Analytical Solution Involving L

The reader may find it useful to derive an analytical solution based on SVD.

Theorem 5. *Let $L = u\mathcal{S}^t v$ be the SVD of L (here $\mathcal{S}$ is indeed square), and let $(u_i)_{i\in\{1,2,\ldots,n\}}$ and $(v_i)_{i\in\{1,2,\ldots,n\}}$ be the columns of the matrices u and v, respectively. The solution for the directed diffusion Equation (52) is given by*

$$f(t,x) = {}^t\alpha^u(t)\cdot u(x) = {}^t\alpha^v(t)\cdot v(x), \tag{55}$$

where

$$\begin{aligned}\begin{pmatrix}\alpha^u(t)\\ \alpha^v(t)\end{pmatrix} &= \begin{pmatrix}\mathrm{ch}(D\,{}^tv\,u\,\mathcal{S}\,t) & -\mathrm{sh}(D\,{}^tv\,u\,\mathcal{S}\,t)\,{}^tvu\\ -\mathrm{sh}(D\,\mathcal{S}\,{}^tv\,u\,t)\,{}^tuv & \mathrm{ch}(D\,\mathcal{S}\,{}^tv\,u\,t)\end{pmatrix}\cdot\begin{pmatrix}\alpha^u(0)\\ \alpha^v(0)\end{pmatrix}\\ &+\int_0^t\begin{pmatrix}\mathrm{ch}(D\,{}^tv\,u\,\mathcal{S}\,(t-\tau)) & -\mathrm{sh}(D\,{}^tv\,u\,\mathcal{S}\,(t-\tau))\,{}^tvu\\ -\mathrm{sh}(D\,\mathcal{S}\,{}^tv\,u\,(t-\tau))\,{}^tuv & \mathrm{ch}(D\,\mathcal{S}\,{}^tv\,u\,(t-\tau))\end{pmatrix}\cdot\begin{pmatrix}\langle\Gamma(\tau,\cdot),U.\rangle_G\\ \langle\Gamma(\tau,\cdot),V.\rangle_G\end{pmatrix}\mathrm{d}\tau\end{aligned}$$

with $\alpha^u(0), \alpha^v(0) \in \mathbb{R}^n$, and

$$\langle\Gamma(\tau,\cdot),u.\rangle_G = \begin{pmatrix}\langle\Gamma(\tau,\cdot),u_1\rangle_G\\ \vdots\\ \langle\Gamma(\tau,\cdot),u_n\rangle_G\end{pmatrix},$$

$$\langle\Gamma(\tau,\cdot),v.\rangle_G = \begin{pmatrix}\langle\Gamma(\tau,\cdot),v_1\rangle_G\\ \vdots\\ \langle\Gamma(\tau,\cdot),v_n\rangle_G\end{pmatrix}.$$

Proof. From the SVD $L = u\,\mathcal{S}\,{}^tv$, we deduce that $L\,{}^tL = u\mathcal{S}\,{}^t\mathcal{S}\,{}^tu$ and ${}^tL\,L = v\,{}^t\mathcal{S}\,\mathcal{S}\,{}^tv$. This means that u and v are the matrices of eigenvectors of the symmetric matrices $L\,{}^tL$ and ${}^tL\,L$, respectively, which thus are diagonalizable matrices. Therefore the columns of u and v form an orthonormal basis of $\mathbb{R}^n$. We can decompose the function f on those two bases:

$$f(t,x) = \sum_{i=1}^n \alpha_i^u(t)\,u_i(x) = \sum_{i=1}^n \alpha_i^v(t)\,v_i(x).$$

Set $i \in \{1,2,\ldots,n\}$. On the one hand, we have

$$\langle f(t,\cdot),\, Lv_i\rangle_G = \langle {}^tLf(t,\cdot),\, v_i\rangle_G$$

The left-hand-side gives

$$\langle f(t,\cdot),\, Lv_i\rangle_G = \langle f(t,\cdot),\, s_iu_i\rangle_G = s_i\,\alpha_i^u(t).$$

The right-hand-side gives, after some calculations

$$D\,\langle {}^tLf(t,\cdot),\, v_i\rangle_G = -\frac{\mathrm{d}}{\mathrm{d}t}\,\langle f(t,\cdot),\, v_i\rangle_G + \langle\Gamma(t,\cdot),\, v_i\rangle_G.$$

This leads to

$$\frac{\mathrm{d}\alpha_i^v}{\mathrm{d}t}(t) = -D\,s_i\,\alpha_i^u(t) + \langle\Gamma(t,\cdot),\, v_i\rangle_G.$$

On the other hand, we have

$$\langle f(t,\cdot),\, {}^tLu_i\rangle_G = \langle Lf(t,\cdot),\, u_i\rangle_G.$$

The left-hand-side gives

$$\langle f(t,\cdot),\, {}^tLu_i\rangle_G = \langle f(t,\cdot),\, s_i v_i\rangle_G = s_i\, \alpha_i^v(t).$$

The right-hand-side gives

$$\langle Lf(t,\cdot),\, u_i\rangle_G = \langle {}^tLf(t,\cdot),\, u_i\rangle_G + \langle (L-{}^tL)\, f(t,\cdot),\, u_i\rangle_G.$$

There is the supplement term $\langle (L-{}^tL)\, f(t,\cdot),\, u_i\rangle_G$. We calculate it below.
We have

$$\langle (L-{}^tL)\, f(t,\cdot),\, u_i\rangle_G = \left\langle \sum_{j=1}^{n} \alpha_j^v(t)\, (L-{}^tL)\, v_j,\, u_i \right\rangle_G = \sum_{j=1}^{n} \alpha_j^v(t) \langle (L-{}^tL)\, v_j,\, u_i\rangle_G.$$

Let $\tilde{\mathcal{S}} = {}^tv\, L\, u = (\tilde{s}_{ij})_{1\le i,j\le n}$. Some calculations lead to

$$\langle (L-{}^tL)\, v_j,\, u_i\rangle_G = {}^tv_j(L-{}^tL)u_i = \tilde{s}_{ji} - s_i\, \delta_{ij}.$$

Finally, we obtain

$$\frac{\mathrm{d}\alpha_i^u}{\mathrm{d}t}(t) = -\sum_{j=1}^{n} D\alpha_j^v(t)\tilde{s}_{ji} + \langle \Gamma(t,\cdot),\, u_i\rangle_G.$$

The system to solve is, therefore,

$$\begin{cases} \dfrac{\mathrm{d}\alpha_i^u}{\mathrm{d}t}(t) & = -\displaystyle\sum_{j=1}^{n} D\alpha_j^v(t)\tilde{s}_{ji} + \langle \Gamma(t,\cdot),\, u_i\rangle_G \\ \dfrac{\mathrm{d}\alpha_i^v}{\mathrm{d}t}(t) & = -D\, s_i\, \alpha_i^u(t) + \langle \Gamma(t,\cdot),\, v_i\rangle_G \end{cases}$$

We are now facing a system of differential equations, first order, non-homogeneous, with constant coefficients. In fact, we can solve this system of equations using a matrix approach.

First, we note that

$$\mathcal{S}\,\tilde{\mathcal{S}} = (\mathcal{S}\,{}^tvu)^2, \quad \tilde{\mathcal{S}}\,\mathcal{S} = ({}^tvu\mathcal{S})^2.$$

Next, we set

$$\mathcal{M} = \begin{pmatrix} 0 & \tilde{\mathcal{S}} \\ \mathcal{S} & 0 \end{pmatrix} \in \mathcal{M}_{2n}(\mathbb{R}),$$

also

$$\gamma(t) = \begin{pmatrix} \langle \Gamma(t,\cdot), u_{\cdot}\rangle_G \\ \langle \Gamma(t,\cdot), v_{\cdot}\rangle_G, \end{pmatrix}$$

and

$$\alpha(t) = \begin{pmatrix} \alpha^u(t) \\ \alpha^v(t) \end{pmatrix} = \begin{pmatrix} \alpha_1^u(t) \\ \alpha_2^u(t) \\ \vdots \\ \alpha_n^u(t) \\ \alpha_1^v(t) \\ \alpha_2^v(t) \\ \vdots \\ \alpha_n^v(t) \end{pmatrix}.$$

The system of differential equations then becomes

$$\frac{\mathrm{d}\alpha}{\mathrm{d}t}(t) = -D\mathcal{M}\alpha(t) + \gamma(t).$$

The Cauchy theorem leads to

$$\alpha(t) = \mathrm{e}^{-D\mathcal{M}t}\,\alpha(0) + \int_0^t \mathrm{e}^{-D\mathcal{M}(t-\tau)}\,\gamma(\tau)\,\mathrm{d}\tau \tag{56}$$

We can go further. We are calculating the powers of the matrix $\mathcal{M}$, as follows. In fact, we can prove by mathematical induction that

$$\mathcal{M}^{2n} = \begin{pmatrix} (\tilde{\mathcal{S}}\mathcal{S})^n & 0 \\ 0 & (\mathcal{S}\tilde{\mathcal{S}})^n \end{pmatrix} \quad \forall n \in \mathbb{N}, \quad \text{and} \quad \mathcal{M}^{2n+1} = \begin{pmatrix} 0 & (\tilde{\mathcal{S}}\mathcal{S})^n\tilde{\mathcal{S}} \\ (\mathcal{S}\tilde{\mathcal{S}})^n\mathcal{S} & 0 \end{pmatrix} \quad \forall n \in \mathbb{N}.$$

Therefore

$$\begin{aligned}
\mathrm{e}^{-D\mathcal{M}t} &= \sum_{l=0}^{+\infty} \frac{(-Dt)^l}{l!}\,\mathcal{M}^l = \sum_{l=0}^{+\infty} \frac{(-Dt)^{2l}}{(2l)!}\,\mathcal{M}^{2l} + \sum_{l=0}^{+\infty} \frac{(-Dt)^{2l+1}}{(2l+1)!}\,\mathcal{M}^{2l+1} \\
&= \begin{pmatrix} \displaystyle\sum_{l=0}^{+\infty} \frac{(-Dt)^{2l}}{(2l)!}\,(\tilde{\mathcal{S}}\mathcal{S})^l & \left(\displaystyle\sum_{l=0}^{+\infty} \frac{(-Dt)^{2l+1}}{(2l+1)!}\,(\tilde{\mathcal{S}}\mathcal{S})^l\right)\tilde{\mathcal{S}} \\ \left(\displaystyle\sum_{l=0}^{+\infty} \frac{(-Dt)^{2l+1}}{(2l+1)!}\,(\mathcal{S}\tilde{\mathcal{S}})^l\right)\mathcal{S} & \displaystyle\sum_{l=0}^{+\infty} \frac{(-Dt)^{2l}}{(2l)!}\,(\mathcal{S}\tilde{\mathcal{S}})^l \end{pmatrix} \\
&= \begin{pmatrix} \displaystyle\sum_{l=0}^{+\infty} \frac{(-Dt)^{2l}}{(2l)!}\,({}^tvu\mathcal{S})^{2l} & \left(\displaystyle\sum_{l=0}^{+\infty} \frac{(-Dt)^{2l+1}}{(2l+1)!}\,({}^tvu\mathcal{S})^{2l}\right)\tilde{\mathcal{S}} \\ \left(\displaystyle\sum_{l=0}^{+\infty} \frac{(-Dt)^{2l+1}}{(2l+1)!}\,(\mathcal{S}\,{}^tvu)^{2l}\right)\mathcal{S} & \displaystyle\sum_{l=0}^{+\infty} \frac{(-Dt)^{2l}}{(2l)!}\,(\mathcal{S}\,{}^tvu)^{2l} \end{pmatrix} \\
&= \begin{pmatrix} \displaystyle\sum_{l=0}^{+\infty} \frac{(-Dt)^{2l}}{(2l)!}\,({}^tvu\mathcal{S})^{2l} & \left(\displaystyle\sum_{l=0}^{+\infty} \frac{(-Dt)^{2l+1}}{(2l+1)!}\,({}^tvu\mathcal{S})^{2l+1}\right){}^tvu \\ \left(\displaystyle\sum_{l=0}^{+\infty} \frac{(-Dt)^{2l+1}}{(2l+1)!}\,(\mathcal{S}\,{}^tvu)^{2l+1}\right){}^tuv & \displaystyle\sum_{l=0}^{+\infty} \frac{(-Dt)^{2l}}{(2l)!}\,(\mathcal{S}\,{}^tvu)^{2l} \end{pmatrix},
\end{aligned}$$

which we can write as

$$\mathrm{e}^{-D\mathcal{M}t} = \begin{pmatrix} \mathrm{ch}(-D\,{}^tv\,u\,\mathcal{S}\,t) & \mathrm{sh}(-D\,{}^tv\,u\,\mathcal{S}\,t)\,{}^tvu \\ \mathrm{sh}(-D\,\mathcal{S}\,{}^tv\,u\,t)\,{}^tuv & \mathrm{ch}(-D\,\mathcal{S}\,{}^tv\,u\,t) \end{pmatrix}.$$

This achieves the proof. □

As a simple verification, what does this theorem become if we set L to be symmetric? We would then have $s_i = \mu_i$, for any $i \in \{1, \ldots, n\}$, and $u = v$, so that ${}^tvu = {}^tuv = \mathrm{Id}$ and $\alpha^u = \alpha^v$. In addition $\tilde{\Sigma}$ would be diagonal and $\tilde{s}_{ii} = \mu_i$, i.e., $\tilde{\mathcal{S}} = \mathcal{S}$. Then

$$\mathrm{ch}(D\,{}^tvu\mathcal{S}t) = \begin{pmatrix} \mathrm{ch}(D\mu_1 t) & & 0 \\ & \ddots & \\ 0 & & \mathrm{ch}(D\mu_n t) \end{pmatrix}$$

and, if $\Gamma = 0$, we have

$$\alpha_i(t) = \mathrm{ch}(D\mu_i t)\alpha_i(0) - \mathrm{sh}(D\mu_i t)\alpha_i(0) = \mathrm{e}^{-D\mu_i t}\alpha_i(0).$$

This conforms to the previous theorem's relevance and truth.

6.3. Solution for the Directed Diffusion Involving the Operator Δ

6.3.1. The Directed Diffusion Equation Involving Δ

Definition 13. *Let $f : E \times \mathbb{R}_+ \to \mathbb{R}$ represent the information function for a graph with n nodes. If D is the diffusion coefficient and $\Gamma : E \times \mathbb{R}_+ \to \mathbb{R}$ the creation function of the information, the following directed diffusion equation holds for any $t \in \mathbb{R}_+$:*

$$\frac{\partial f}{\partial t}(t,x) = -D\,{}^t\Delta\, f(t,x) + \Gamma(t,x) \tag{57}$$

We have the same Equation (22) if and only if $\Delta = {}^t\Delta$, that is if and only if A is symmetrical.

6.3.2. Solution Involving Δ

Theorem 6 (Application of the Cauchy Theorem)**.** *The solution of the related Cauchy problem associated with Equation (13) is given by*

$$f(t,x) = \mathrm{e}^{-D\,{}^t\Delta\, t}\, f(0,x) + \int_0^t \mathrm{e}^{-D\,{}^t\Delta\,(t-\tau)}\,\Gamma(\tau,x)\,\mathrm{d}\tau. \tag{58}$$

This expression is straightforwardly implementable in a computer.

6.3.3. Analytical Solution Involving Δ

Similarly, we can prove the following theorem.

Theorem 7. *Let $\tilde{\Delta} = U\,\Sigma\,{}^tV$ be the SVD of $\tilde{\Delta}$ (here Σ is indeed square), and let $(U_i)_{i\in\{1,2,\ldots,n\}}$ and $(V_i)_{i\in\{1,2,\ldots,n\}}$ be the columns of the matrices U and V, respectively. The solution for the directed diffusion Equation (57) is given by*

$$f(t,x) = \frac{1}{\sqrt{\mathrm{d}_A x}}\,{}^t\beta^u(t)\cdot U(x) = \frac{1}{\sqrt{\mathrm{d}_A x}}\,{}^t\beta^v(t)\cdot V(x), \tag{59}$$

where

$$\begin{aligned}\begin{pmatrix}\beta^u(t)\\ \beta^v(t)\end{pmatrix} &= \begin{pmatrix}\mathrm{ch}(D\,{}^tV\,U\,\mathcal{S}\,t) & -\mathrm{sh}(D\,{}^tV\,U\,\mathcal{S}\,t)\,{}^tVU\\ -\mathrm{sh}(D\,\mathcal{S}\,{}^tV\,U\,t)\,{}^tUV & \mathrm{ch}(D\,\mathcal{S}\,{}^tV\,U\,t)\end{pmatrix}\cdot\begin{pmatrix}\beta^u(0)\\ \beta^v(0)\end{pmatrix}\\ &+\int_0^t \begin{pmatrix}\mathrm{ch}(D\,{}^tV\,U\,\mathcal{S}\,(t-\tau)) & -\mathrm{sh}(D\,{}^tV\,U\,\mathcal{S}\,(t-\tau))\,{}^tVU\\ -\mathrm{sh}(D\,\mathcal{S}\,{}^tV\,U\,(t-\tau))\,{}^tUV & \mathrm{ch}(D\,\mathcal{S}\,{}^tV\,U\,(t-\tau))\end{pmatrix}\cdot\begin{pmatrix}\langle\mathcal{D}^{1/2}\,\Gamma(\tau,\cdot),U_\cdot\rangle_G\\ \langle\mathcal{D}^{1/2}\,\Gamma(\tau,\cdot),V_\cdot\rangle_G\end{pmatrix}\mathrm{d}\tau\end{aligned}$$

with $\beta^u(0),\beta^v(0)\in\mathbb{R}^n$, and

$$\langle\mathcal{D}^{1/2}\,\Gamma(\tau,\cdot),U_\cdot\rangle_G = \begin{pmatrix}\langle\mathcal{D}^{1/2}\,\Gamma(\tau,\cdot),U_1\rangle_G\\ \vdots\\ \langle\mathcal{D}^{1/2}\,\Gamma(\tau,\cdot),U_n\rangle_G\end{pmatrix},$$

$$\langle\mathcal{D}^{1/2}\,\Gamma(\tau,\cdot),V_\cdot\rangle_G = \begin{pmatrix}\langle\mathcal{D}^{1/2}\,\Gamma(\tau,\cdot),V_1\rangle_G\\ \vdots\\ \langle\mathcal{D}^{1/2}\,\Gamma(\tau,\cdot),V_n\rangle_G\end{pmatrix}.$$

7. Directed Graph: Case Studies

This section shows simulations for important graphs.

7.1. Erdos–Reyni

The Erdos–Reyni graph is uniformly random: fix the number of nodes and edges. Then the connections are all random. Figure 5 shows a simulation.

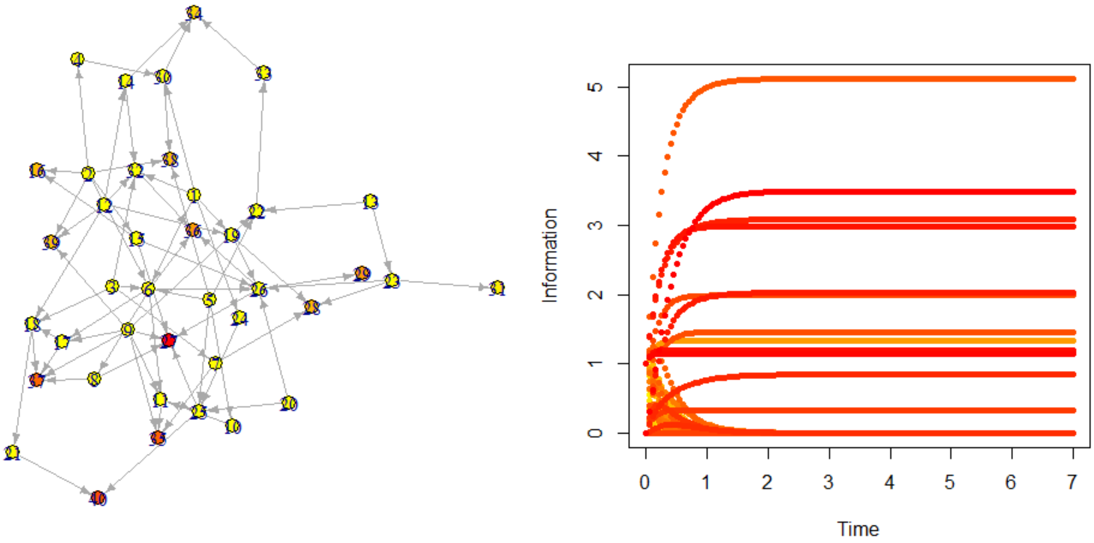

Figure 5. Diffusion on a directed Erdos–Reyni graph.

There might be counter-intuitive facts, e.g., the node which has a high number of neighbors is not necessarily the most influential in the network. The influence depends on connectivity, but also on a given network configuration, as distribution of information among all the nodes.

7.2. Binary Tree

The binary graph is the simplest hierarchical structure on graphs: every node has two children. Figure 6 shows a simulation.

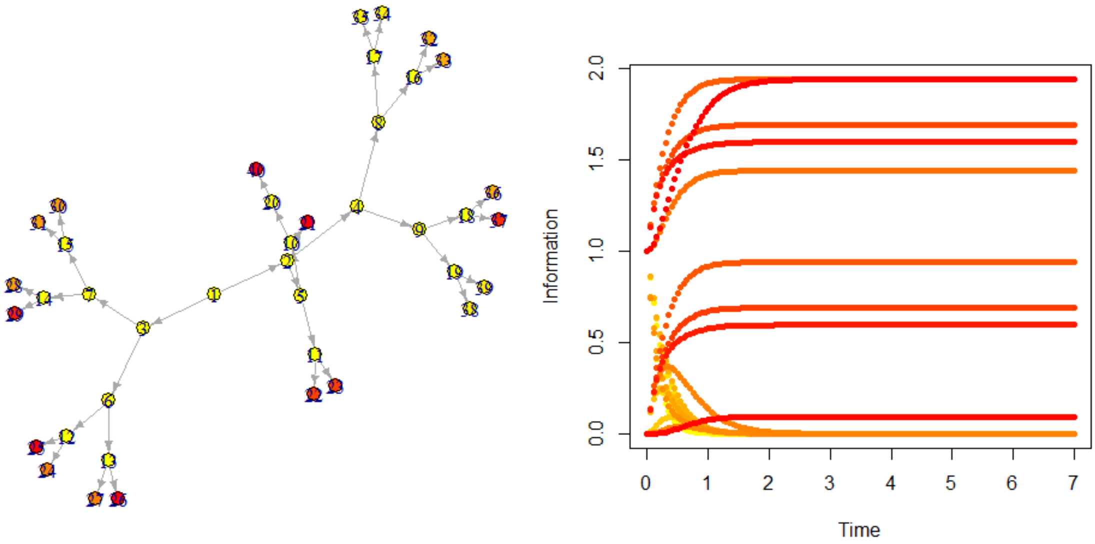

Figure 6. Diffusion on a directed binary tree.

In the standard configuration as in Figure 6, the information is circulating from parents to children. We, however, observed (as in Mandala networks) the other possible direction.

7.3. 4-Tree

Figure 7 shows a simulation.

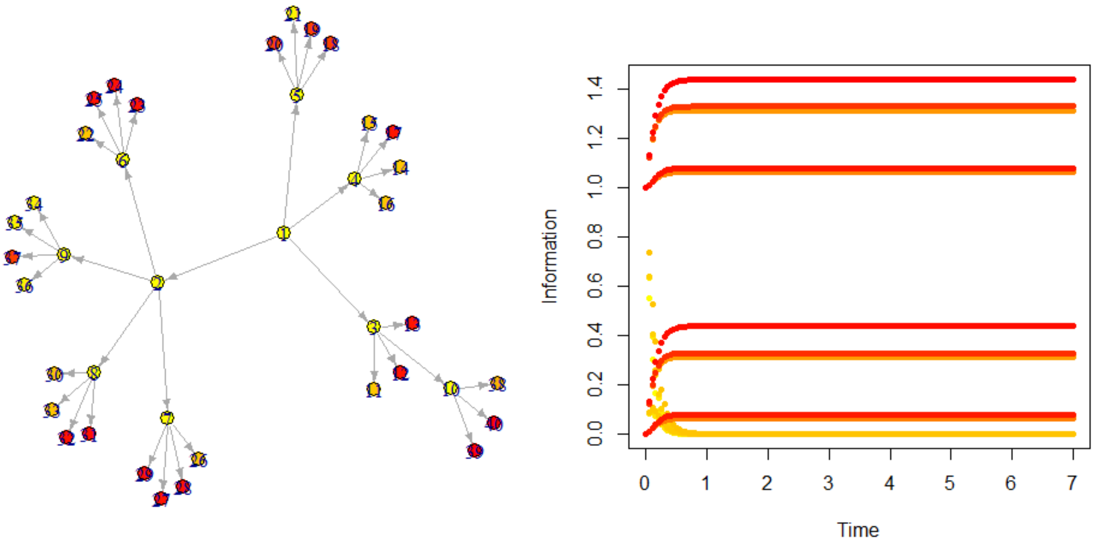

Figure 7. Diffusion on a 4-tree.

7.4. Ring Graph

The ring graph is the simplest cycle on a network. Cycles are a very important aspect of graphs: they allow to extract seasonality patterns for a given context. Figure 8 shows a simulation.

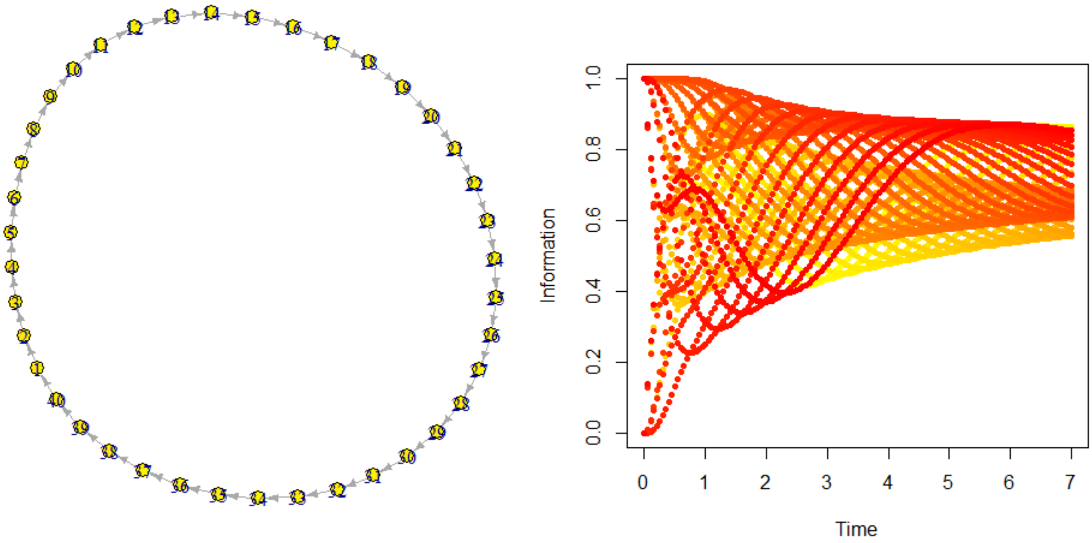

Figure 8. Diffusion on a directed ring.

The most important characteristics we observe is the envelope: the cycle behaves as two distinct sub-systems which interact continuously up to the dissipation level. One sub-system gives the information to the other, which receives it, and vice versa.

8. Layers Constantly Giving Information

8.1. Motivations

This section deals with an important aspect of the Bitcoin network. There are different families (layers) of nodes in this network. In fact, the P2P network is a subnetwork of the whole system. There is, in addition, the layer of *users* which may send information (e.g., a transaction) to the miners so that they mine it. There also is the layer of *exchanges*, which sends in the blockchain any information which may be about public key diffusion, or even collects prices from different exchanges. In particular, an exchange needs to be in the center of the whole network, communicating to nodes of all the layers by sending and receiving all types of information. Such a complex system could in fact be simplified and illustrated in Figure 9. If the related graph is directed, then the layers are said to be *directed*, and the layer which receives information from others, and whose nodes exchange information with each other, is named the *main layer*. A graph configuration where the main layer is the layer of miners could represent all the list of transactions diffusion from users to miners, and also between miners (see Figure 9), whereas if the main layer is the layer of exchanges, the resulting diffusion model would describe all the interactions (i.e., emission/reception) of data from users, exchanges, and miners, the exchanges could have. In particular, this graphical representation of an exchange should genuinely reveal, in a clear way, the whole flow of information dealing with the exchange. The diffusion model, for a given time condition, should predict the evolution of information propagation all around the network. Modeling this is the object of the current section. Surprisingly, bearing in mind the previous section, the result is simple.

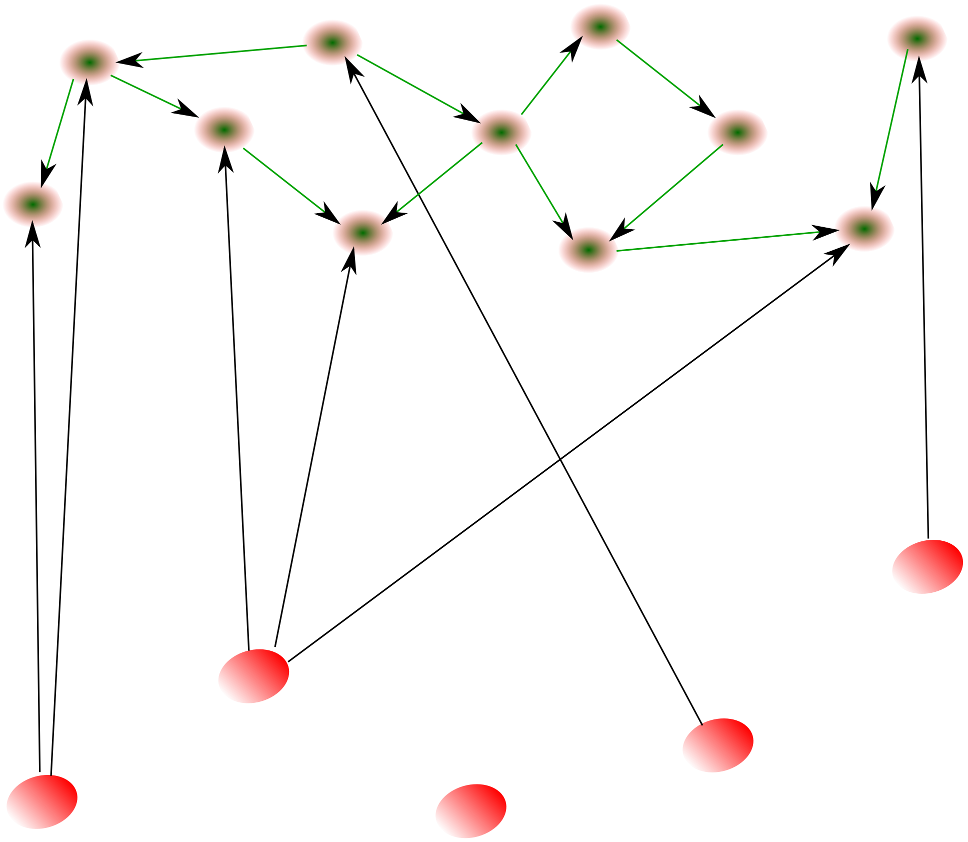

Figure 9. Two directed layers, i.e., families of nodes. The red layer sends information toward the green layer, while nodes of the last layer send information to each other. Thus, the green layer could represent the layer of miners, while the red layer could represent the layer of exchanges. In such a model, the miners are the main layer. We could easily change the configuration so that exchanges are the main layer, and the following theory also applies.

8.2. Directed Diffusion with Boundary Conditions—Analytical Expression

Let $S \subset G$ be the main layer (the green layer in Figure 9), and $\bar{S} \subset G$ (the red layer in Figure 9) such that $S \cup \bar{S} = G$ and $S \cap \bar{S} = \emptyset$. We keep the notations in the previous sections. The following theorem is the Theorem 5, with an additional boundary condition.

Theorem 8. *Suppose L is written as*

$$L = \begin{pmatrix} L_S & * \\ * & L_{\bar{S}} \end{pmatrix}, \tag{60}$$

where, in particular, L_S is the Laplacian of the sub-graph S with singular vectors u^S and v^S. Suppose $f(t,x) = B(t,x)$, fixed and known, for any $(t,x) \in \mathbb{R}_+ \times \bar{S}$ (boundary condition). Then, Theorem 5 applies to the sub-graph S, by replacing $\Gamma(\tau,\cdot), u_.\rangle_G$ with $\langle\tilde{\Gamma}(\tau,\cdot), u^S_.\rangle_S$, and $\langle\Gamma(\tau,\cdot), v_.\rangle_G$ with $\langle\tilde{\Gamma}(\tau,\cdot), v^S_.\rangle_S$, where

$$\tilde{\Gamma}(t,x) = \Gamma(t,x) + D \sum_{y\in\bar{S}} B(t,y)\, a_{yx} \tag{61}$$

Proof. Let $\tilde{\mu}_1 \geq \ldots \geq \tilde{\mu}_{s-1} > \tilde{\mu}_s = 0$ be the singular values for L_S, and let u^S and v^S be the singular vector matrices for L_S. In fact, we set $u^S_i(x) = 0$ if $x \notin S$, idem for v^S, for any $i \in \{1,2,\ldots,n\}$, so that u^S_i and v^S_i are n-dimensional vectors.

Bearing in mind the proof for Theorem 5, we need to set

$$f_{|S}(t,x) = \sum_{i=1}^{n} a^u_i(t)\, u^S_i(x) = \sum_{i=1}^{n} a^v_i(t)\, v^S_i(x), \quad \forall (t,x) \in \mathbb{R}_+ \times S.$$

On the one hand, we have

$$\begin{aligned}
D\, \tilde{\mu}_j\, a^u_j(t) &= D\left\langle f_{|S}(t,\cdot),\, \tilde{\mu}_j\, u^S_j \right\rangle_S = D\left\langle f_{|S}(t,\cdot),\, L_S\, v^S_j \right\rangle_S \\
&= D\left\langle f(t,\cdot),\, L\, v^S_j \right\rangle_G - D\left\langle f(t,\cdot),\, L\, v^S_j \right\rangle_{\bar{S}} \\
&= \left\langle D\, {}^t L f(t,\cdot),\, v^S_j \right\rangle_G - D \sum_{y\in\bar{S}} f(t,y) \left(L\, v^S_j\right)(y) \\
&= -\frac{\mathrm{d}}{\mathrm{d}t} a^v_j(t) + \left\langle \Gamma(t,\cdot),\, v^S_j \right\rangle_G - D \sum_{x\in G} \sum_{y\in\bar{S}} f(t,y) \left(v^S_j(y) - v^S_j(x)\right) a_{yx} \\
&= -\frac{\mathrm{d}}{\mathrm{d}t} a^v_j(t) + \left\langle \Gamma(t,\cdot),\, v^S_j \right\rangle_G + D \sum_{x\in S} \sum_{y\in\bar{S}} B(t,y)\, v^S_j(x)\, a_{yx}.
\end{aligned}$$

Thus

$$D\, \tilde{\mu}_j\, a^u_j(t) = -\frac{\mathrm{d}}{\mathrm{d}t} a^v_j(t) + \left\langle \Gamma(t,\cdot) + D \sum_{y\in\bar{S}} B(t,y)\, a_{y\,\cdot},\, v^S_j \right\rangle_G .$$

Identically, we have

$$\begin{aligned}
D\, \tilde{\mu}_j\, a^v_j(t) &= D\left\langle f_{|S}(t,\cdot),\, {}^t L_S\, u^S_j \right\rangle_S \\
&= D\left\langle {}^t L\, f(t,\cdot),\, u^S_j \right\rangle_G + D\left\langle (L - {}^t L)\, f(t,\cdot),\, u^S_j \right\rangle_G - D\left\langle f(t,\cdot),\, {}^t L\, u^S_j \right\rangle_{\bar{S}}.
\end{aligned}$$

In particular, the last term is written as

$$
\begin{aligned}
-D\left\langle f(t,\cdot),\, {}^tL\, u_j^S \right\rangle_{\bar{S}} &= -D \sum_{y\in\bar{S}} f(t,y) \sum_{x\in G} \left(u_j^S(y)\, {}^ta_{yx} - u_j^S(x)\, {}^ta_{xy} \right) \\
&= -D \sum_{x\in G} \sum_{y\in\bar{S}} B(t,y) \left(u_j^S(y)\, {}^ta_{yx} - u_j^S(x)\, {}^ta_{xy} \right) \\
&= D \sum_{x\in S} \sum_{y\in\bar{S}} B(t,y)\, u_j^S(x)\, {}^ta_{xy} \\
&= D \left\langle \sum_{y\in\bar{S}} B(t,y)\, a_{y\cdot},\, u_j^S \right\rangle_S .
\end{aligned}
$$

The same calculations performed as in the proof for Theorem 5 actually allow to conclude. □

Remark 4. *When the exchanges constitute the main layer S, two other layers could be of interest, the layers of users (customers) $\bar{S}_1$ and the layers of miners $\bar{S}_2$, so that the whole network G is such that $G = S \cup \bar{S}_1 \cup \bar{S}_2$, and $S \cap \bar{S}_1 \cap \bar{S}_2 = \emptyset$. Setting $\bar{S} = \bar{S}_1 \cup \bar{S}_2$, we deduce that Theorem 8 applies to the layer S of exchanges, and, considering the two distinct sources $\{\text{customers}, \text{miners}\}$, we specifically have*

$$
\tilde{\Gamma}(t,x) = \Gamma(t,x) + D \sum_{y\in\bar{S}_1} B_1(t,y)\, a_{yx} + D \sum_{y\in\bar{S}_2} B_2(t,y)\, a_{yx}. \tag{62}
$$

8.3. Directed Diffusion with Boundary Conditions—Practical Case

The previous section demonstrates that any boundary condition must be considered as a constraint to the nodes whose neighbors are boundary nodes, multiplied by the diffusion coefficient D. Figure 9 thus becomes the object of Figure 10.

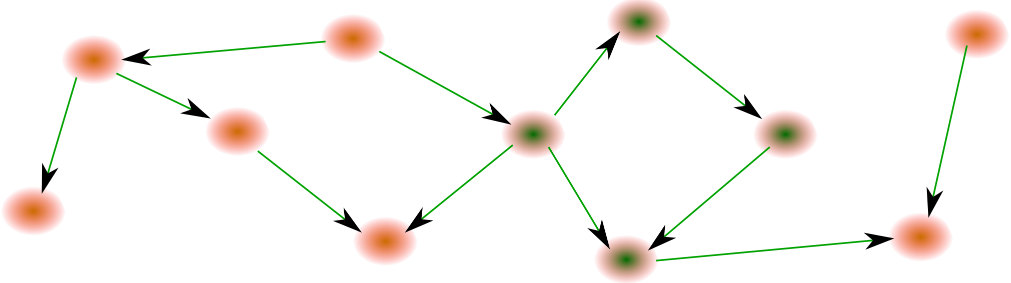

Figure 10. Provided that the red-nodes layer plays the role of boundary condition layer in Figure 9, the two directed layers network is mathematically equivalent to one unique network, whose node neighbors with the boundary layer nodes have a reaction term Γ applied for the diffusion of information.

Basically, each contrainst node has a stronger diffusive effect toward the rest of the network. More specifically, since (supposing Δt sufficiently small)

$$
\int_t^{t+\Delta t} \mathrm{e}^{-D\,{}^tL\,(t-\tau)}\, \Gamma(\tau,x)\, \mathrm{d}\tau = \Gamma(t,x)\, \Delta t + \mathrm{o}(\Delta t). \tag{63}
$$

and from Equation (54), we can set

$$
f(t+\Delta t,x) = \mathrm{e}^{-D\,{}^tL\,\Delta t} f(t,x) + \Gamma(t,x)\, \Delta t + \mathcal{E}(t,x), \tag{64}
$$

where $\mathcal{E}(t,x) = \mathrm{o}(\Delta t)$ is the error committed by the approximation. In practice, the approximation is sufficiently good if $\Delta t \leq 0.05$ second such that we set $\Delta t = 0.05$. Provided that Γ is of class $\mathcal{C}^\infty$, we indeed have an error of

$$\begin{aligned}\mathcal{E}(t,x) &= \int_t^{t+\Delta t} \mathrm{e}^{-D\,{}^tL\,(t-\tau)}\,\Gamma(\tau,x)\,\mathrm{d}\tau - \Gamma(t,x)\,\Delta t \\ &\leq \left|\sum_{i=2}^{+\infty}\frac{\Delta t^n}{n!}\Gamma^{(n-1)}(t,x)\right| \leq \sum_{i=2}^{+\infty}\frac{\Delta t^n}{n!}\left|\Gamma^{(n-1)}(t,x)\right|.\end{aligned}$$

If the function Γ is sufficiently smooth, then

$$\mathcal{E}(t,x) \leq \left(\sum_{i=2}^{+\infty}\frac{\Delta t^n}{n!}\right)\sup_{n\in\mathbb{N}^*}\left(\left|\Gamma^{(n)}(t,x)\right|\right) = \left(\mathrm{e}^{\Delta t} - 1 - \Delta t\right)\sup_{n\in\mathbb{N}^*}\left(\left|\Gamma^{(n)}(t,x)\right|\right).$$

In practice, we shall have $\sup\limits_{n\in\mathbb{N}^*}\left(\left|\Gamma^{(n)}(t,x)\right|\right) \leq 1$ so that the error $\mathcal{E}$ never exceeds 2% such that the equality $\mathcal{E}(t,x) = \mathrm{o}(\Delta t)$ is justified, for any x.

Figure 11 plots a diffusion without the creation/destruction ($\Gamma = 0$) of a graph, where the information is provided by node 1, and diffuses up to node 10 (without ever reaching 11). Note that nodes 2 and 3 (respectively, 5, 6, and 7) play symmetrical roles and have the same information amount through time.

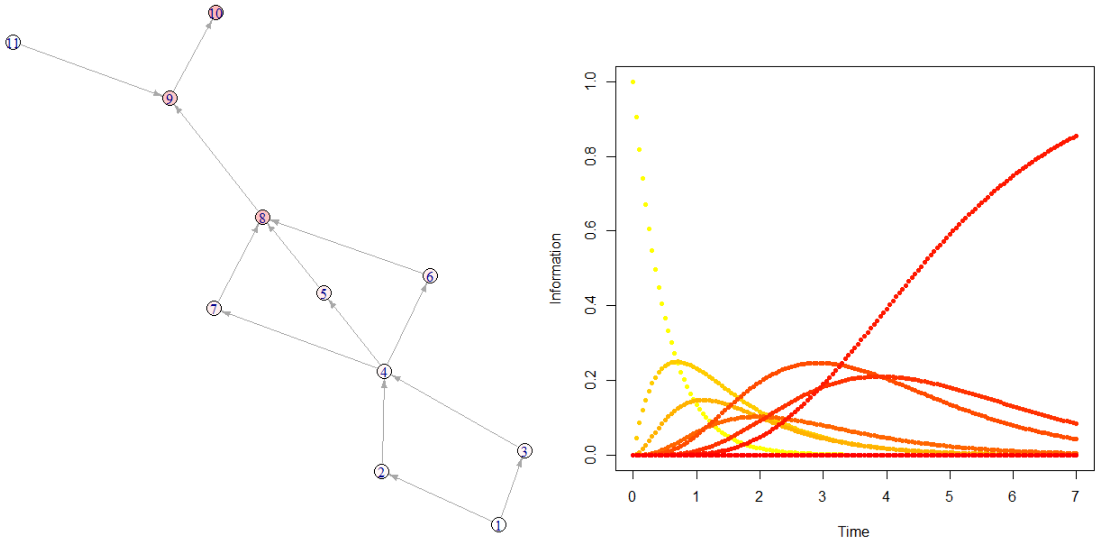

Figure 11. The information diffuses from node 1 ($f(0,1) = 1$ while $f(0,x) = 0$ for any $x \neq 1$) to node 10. Here, $\Gamma = 0$ and $D = 1$.

We now add the constraint Γ such that

$$\Gamma(t,x) = \begin{cases} \Gamma(t-\Delta t, x) + \dfrac{r}{\Delta t} & \text{if } x = 2 \quad \text{and } f(t,2) \leq 1 \\[2ex] \Gamma(t-\Delta t, x) - \dfrac{r'}{\Delta t} & \text{if } x = 10 \quad \text{and } f(t,10) \geq 1 \\[2ex] 0 & \text{otherwise} \end{cases}$$

In words, node 2 is constantly fed by external layers with a rate of r (creation) if it does not have a sufficient resource, while node 10 is constantly feeding external layers with a rate of r' (destruction) if it does have too much of a resource. We choose rates $r = r' = \Delta t = 0.05$.

With a cubic spline smoothing, we may observe that $\sup_{n \in \mathbb{N}^*} \left(\left| \Gamma^{(n)}(t,x) \right| \right) \leq 1$ so that $\mathcal{E} \leq 0.2\%$.

Figure 12 shows the simulation.

It is interesting to note the literal change of behavior by applying such constraints. Node 2 is submitted to permanent oscillations around 1 with a frequency-decreasing transition regime, so node 10 once it reaches information higher than 1. The oscillations induced at node 2 transfer visible oscillations to node 4, but the diffusion is not sufficiently strong to see them.

We therefore increase the diffusion coefficient D from 1 to 5, then to 10, and Figure 13 shows the simulations.

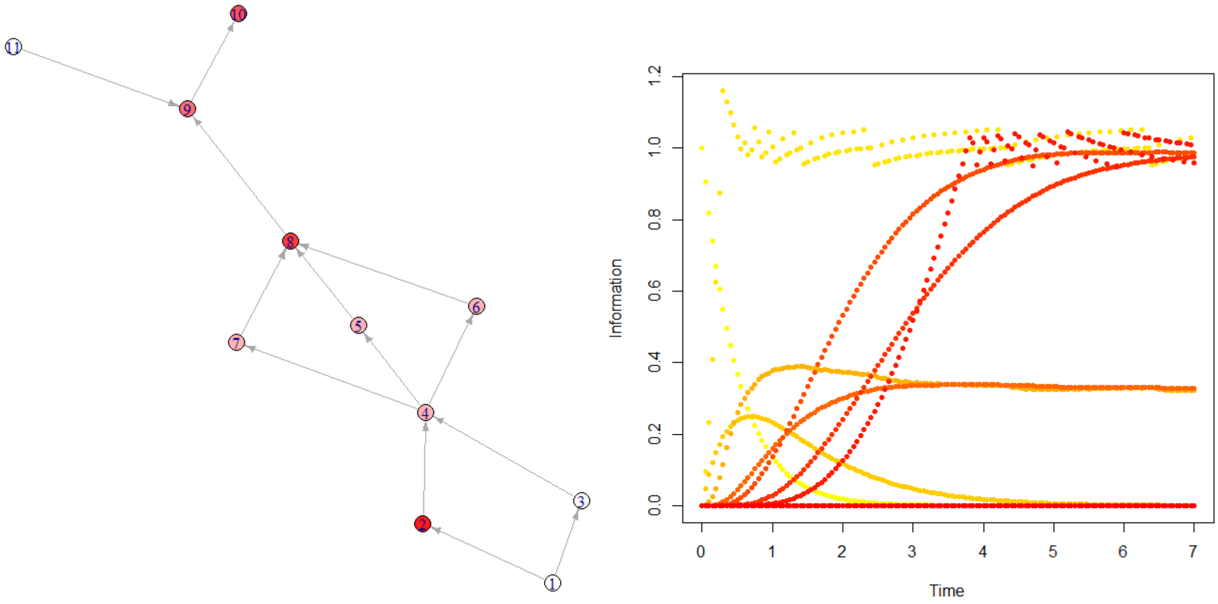

Figure 12. The information diffuses from node 1 ($f(0,1) = 1$ while $f(0,x) = 0$ for any $x \neq 1$) to node 10 with additional creation (to node 2) and destruction (to node 10) added. Here $D = 1$.

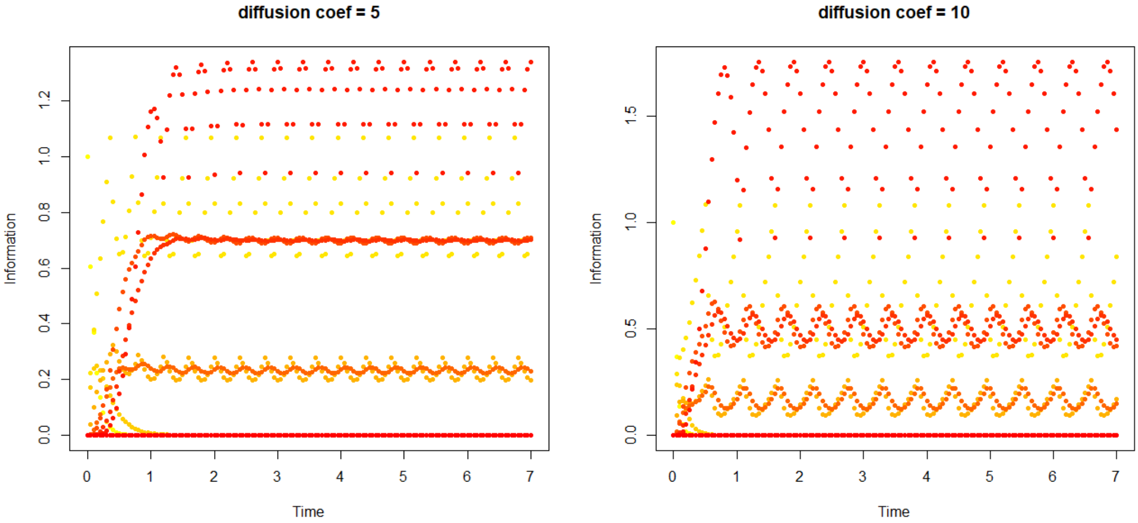

Figure 13. The information diffuses from node 1 ($f(0,1) = 1$ while $f(0,x) = 0$ for any $x \neq 1$) to node 10 with additional creation (to node 2) and destruction (to node 10) added. Here $D = 5$ (**left** panel) and $D = 10$ (**right** panel).

For all intermediate nodes, oscillations of information are observed at a frequency identical to the one imposed by node 2 reception. Concretely, it means that one node providing constant information to the others without exceeding its resources is governing the dynamics of the network. The shift of the minimum and maximum values at the permanent regime is due to the *delay* to obtain the information, and the shift is higher if the node is more distant from node 2.

In fact, the graph in Figures 12 and 13 behaves as a cycle: nodes 2 and 10 could be connected to each other through an intermediate set of layers. This means that cycles are maintaining oscillations through a permanent regime. These oscillations are importantly induced by one parameter: the diffusion coefficient D, which therefore appears as a *bifurcation parameter* (its value determines chaotic behavior in the network).

9. Delay in the Differential Equation

So far, there has been an immediate response from a stimulation of the network. However, time lags are essential to consider since response to the information never is automatic: there is a response time to understand the information and find an appropriate answer, from the human as well as machine viewpoints. It is essential to find a way to describe delay in the study.

Generally speaking, a delayed differential equation—an active topic of academic research nowadays—for the above diffusion equation could be written as a delayed-response diffusion equation:

$$\frac{\partial f}{\partial t}(t,x) = -D\,{}^{t}L\,f(t-\tau,x) + \Gamma(t,x), \tag{65}$$

where $\tau \geq 0$ is the *time lag* for the answer of the system due to diffusion effects. Usually, the function $f(\cdot,x)$, for any $x \in G$, is defined on $[-\tau,+\infty[$ and is given on $[-\tau,0]$.

Although there are methods for solving such an equation, we may develop some technics to make an easy implementation in the following. One trick is to make L time dependent, as we are going to see.

9.1. Time Dependent Laplacian $L(t)$

The adjacency matrix may be time dependent, as nodes can change their configuration through time. In the concern of expressing the time-dependent Laplacian matrix $L(t) = \mathcal{D}(t) - A(t)$ with respect to $A(t)$ exclusively, we note that

$$\mathcal{D}(t) = \sum_{i,j=1}^{n} E_{i,i}\,A(t)\,E_{j,i},$$

and the Laplacian matrix is rewritten as $L(t) = \sum_{i,j=1}^{n} E_{i,i}\,A(t)\,E_{j,i} - A(t)$, which behaves as a functional of the matrix $A(t)$.

Bearing this in mind, a way to introduce delay in a graph system is to make connections at a given time $\tau > 0$. Without any loss of generality, we suppose that node rth, for a fixed $r \in [\![1,n]\!]$, is an isolated node at time $t \in [0,\tau[$, then gets connected to, say, $k \in [\![1,n]\!] \setminus \{r\}$ (from k to r) in the network, from time $t = \tau$.

This makes the adjacency matrix $A(t)$ time dependent, since

$$a_{kr}(t) = \begin{cases} 0 & \text{if } t \in [0,\tau[\\ 1 & \text{if } t \in [\tau,+\infty[\end{cases} = \mathrm{H}(t-\tau),$$

where H is the Heaviside distribution.

Thus, we can prove that the rth diagonal element of L is $l_{rr}(t) = \sum_{i=1}^{n} a_{ri} = l_{rr}(0) + \mathrm{H}(t-\tau)$, while $l_{kr}(t) = -a_{kr}(t) = -\mathrm{H}(t-\tau)$. In this configuration, we therefore have

$$L(t) = L(0) + \mathcal{H}(t)$$

where $\mathcal{H}(t)$ is zero everywhere, except at element (k,r), where it is equal to $-\mathrm{H}(t-\tau)$, and at its diagonal element (r,r), which is equal to $\mathrm{H}(t-\tau)$. We actually can generalize the above approach to the following proposition.

Proposition 7. *If $p \in \mathbb{N}^*$ nodes $x_1, x_2, \ldots, x_p$ are connected to $x_{11}, \ldots, x_{1q_1}, \ldots$, and $x_{p1}, \ldots, x_{pq_p}$, respectively, at times $\tau_{11}, \ldots, \tau_{1q_1}, \ldots$, and $\tau_{p1}, \ldots, \tau_{pq_p}$, respectively, then*

$$L(t) = L(0) + \mathcal{H}(t), \tag{66}$$

where $\mathcal{H}(t)$ is zero except element (x_i, x_i) which is $\sum_{k=1}^{q_i} \mathrm{H}(t-\tau_{ik})$, for all $i \in [\![1,p]\!]$, and either element (x_{ik}, x_i) if the connection is from x_{ik} to x_i, or element (x_i, x_{ik}) otherwise, whose value is $-\mathrm{H}(t-\tau_{ik})$, for all $i \in [\![1,p]\!]$, $k \in [\![1,q_i]\!]$.

Thus, the matrix $\mathcal{H}$ behaves as a Laplacian matrix whose elements are either 0 or Heaviside distributions.

The idea now is to find a solution with the Heaviside distribution as a kernel.

9.2. Definition Set for the Solution and Its Derivative

From the previous Section 9.1, the diffusion equation is written as

$$\frac{\partial f}{\partial t}(t,x) = -D^{\,t}(L(0) + \mathcal{H}(t))\, f(t,x) + \Gamma(t,x).$$

It is worth stressing that the matrix $\mathcal{H}(t)$ is not defined at times when an evenness occurs, which therefore implies that f is not derivable on those points. It turns out that f is of class $\mathcal{C}$ almost everywhere (and not everywhere), provided that Γ is continuous almost everywhere as well, but still is required to be continuous at all times. If the study starts at $t = 0$ and there is no connection between distant nodes, then $L(0) = 0_{\mathcal{M}_n(\mathbb{R})}$ and the equation simplifies to

$$\frac{\partial f}{\partial t}(t,x) = -D^{\,t}\mathcal{H}(t)\, f(t,x) + \Gamma(t,x),$$

which has the usual form.

9.3. Simple Cases

9.3.1. Heaviside integration

Suppose we would like to calculate

$$I = \int \mathrm{H}(t-\tau)\, g(t)\, \mathrm{d}t,$$

where g is a continuous and integrable function. Let

$$G(t) = \int g(t)\, \mathrm{d}t,$$

up to a constant. Then

$$I = \int \mathrm{H}(t-\tau)\, \mathrm{d}G(t).$$

By integration by parts, we have

$$\begin{aligned} I =&\ \mathrm{H}(t-\tau)\, G(t) - \int G(t)\, \mathrm{dH}(t-\tau) \\ =&\ \mathrm{H}(t-\tau)\, G(t) - \int G(t)\, \delta(t-\tau)\, \mathrm{d}t \\ =&\ \mathrm{H}(t-\tau)\, G(t) - \mathrm{H}(t-\tau)\, G(\tau), \end{aligned}$$

true up to a constant $\mathrm{C^{te}} \in \mathbb{R}$, and we thus can write

$$\int \mathrm{H}(t-\tau)\, g(t)\, \mathrm{d}t = \mathrm{H}(t-\tau)\, (G(t) - G(\tau)) + \mathrm{C^{te}}. \tag{67}$$

9.3.2. Simple Case of Two Nodes

Suppose in $E = \{1, 2\}$, the network is composed of two nodes. Without creation/destruction, suppose that node 1 sends information to node 2 from time $\tau > 0$. The Laplacian matrix is written as

$$L = \begin{pmatrix} \mathrm{H}(t-\tau) & -\mathrm{H}(t-\tau) \\ 0 & 0 \end{pmatrix}. \tag{68}$$

This equation is just from the diffusion equations given by the system:

$$\begin{cases} \dfrac{\mathrm{d}f_1}{\mathrm{d}t}(t) &= -D\, \mathrm{H}(t-\tau)\, f_1(t) \\ \dfrac{\mathrm{d}f_2}{\mathrm{d}t}(t) &= \ \ D\, \mathrm{H}(t-\tau)\, f_1(t) \end{cases}$$

According to Equation (67) with $g = 1$, the first equation gives

$$\ln f_1(t) = \ln f_1(\tau) - D\,(t-\tau)\, \mathrm{H}(t-\tau),$$

or

$$f_1(t) = f_1(\tau)\, \mathrm{e}^{-D\,(t-\tau)\,\mathrm{H}(t-\tau)}.$$

Note that we can rewrite this function as

$$\begin{aligned} f_1(t) =& f_1(\tau)\,(1 - \mathrm{H}(t-\tau)) + f_1(\tau)\, \mathrm{e}^{-D\,(t-\tau)}\, \mathrm{H}(t-\tau) \\ =& f_1(\tau)\, \Big(1 - (1 - \mathrm{e}^{-D\,(t-\tau)})\, \mathrm{H}(t-\tau)\Big). \end{aligned}$$

Then putting this expression into the second equation and using the fact that $\mathrm{H}(t-\tau)^2 = \mathrm{H}(t-\tau)$, we have

$$\frac{\mathrm{d}f_2}{\mathrm{d}t}(t) = D\, \mathrm{H}(t-\tau)\, \mathrm{e}^{-D\,(t-\tau)}.$$

Using again Equation (67) with $g(t) = \mathrm{e}^{-D\,(t-\tau)}$, we have

$$f_2(t) = f_2(\tau) + f_1(\tau)\,(1 - \mathrm{e}^{-D\,(t-\tau)})\, \mathrm{H}(t-\tau).$$

This behavior is the one expected.

9.3.3. Simple Case of Three Nodes

The case of three nodes $E = \{1, 2, 3\}$ might be of particular interest as well. We could break the symmetry by supposing that node 1 sends information to node 2 at rate α from time $\tau > 0$, and to node 3 at rate β from time $\tau' \geq \tau$. The Laplacian matrix is written as

$$L = \begin{pmatrix} \alpha \mathrm{H}(t-\tau) + \beta \mathrm{H}(t-\tau') & -\alpha \mathrm{H}(t-\tau) & -\beta \mathrm{H}(t-\tau') \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}. \tag{69}$$

This equation is just from the diffusion equations given by the system:

$$\begin{cases} \dfrac{\mathrm{d}f_1}{\mathrm{d}t}(t) & = -D\,(\alpha \mathrm{H}(t-\tau) + \beta \mathrm{H}(t-\tau'))\, f_1(t) \\ \dfrac{\mathrm{d}f_2}{\mathrm{d}t}(t) & = \quad D\,\alpha \mathrm{H}(t-\tau)\, f_1(t) \\ \dfrac{\mathrm{d}f_3}{\mathrm{d}t}(t) & = \quad D\,\beta \mathrm{H}(t-\tau')\, f_1(t) \end{cases}$$

Again, according to Equation (67), the first equation gives

$$\ln f_1(t) = \ln f_1(\tau) - D\,(t-\tau)\,\alpha \mathrm{H}(t-\tau) - D\,(t-\tau')\,\beta \mathrm{H}(t-\tau'),$$

or

$$f_1(t) = f_1(\tau)\, \mathrm{e}^{-D\,((t-\tau)\,\alpha \mathrm{H}(t-\tau) + (t-\tau')\,\beta \mathrm{H}(t-\tau'))},$$

which we can rewrite, since $f_1(\tau) = f_1(0)$, as

$$f_1(t) = f_1(0)\left(1 - (1 - \mathrm{e}^{-D\,\alpha\,(t-\tau)})\, \mathrm{H}(t-\tau)\right)\left(1 - (1 - \mathrm{e}^{-D\,\beta\,(t-\tau')})\, \mathrm{H}(t-\tau')\right).$$

Deriving f_2 is the most complex task here. First of all, we note that $\mathrm{H}(t-\tau)\mathrm{H}(t-\tau') = \mathrm{H}(t-\tau')$, and then we have

$$\begin{aligned} \frac{\mathrm{d}f_2}{\mathrm{d}t}(t) &= D\,\alpha \mathrm{H}(t-\tau)\, f_1(0)\left(1 - (1 - \mathrm{e}^{-D\,\alpha\,(t-\tau)})\, \mathrm{H}(t-\tau)\right) \\ &\quad \times \left(1 - (1 - \mathrm{e}^{-D\,\beta\,(t-\tau')})\, \mathrm{H}(t-\tau')\right) \\ &= D\,\alpha \mathrm{H}(t-\tau)\, f_1(0)\, \mathrm{e}^{-D\,\alpha\,(t-\tau)} \\ &\quad \times \left(1 - (1 - \mathrm{e}^{-D\,\beta\,(t-\tau')})\, \mathrm{H}(t-\tau')\right) \\ &= D\,\alpha\, f_1(0)\left(\mathrm{e}^{-D\,\alpha\,(t-\tau)}\, \mathrm{H}(t-\tau)\right. \\ &\quad \left. -(\mathrm{e}^{-D\,\alpha\,(t-\tau)} - \mathrm{e}^{-D[\alpha\,(t-\tau) + \beta\,(t-\tau')]})\, \mathrm{H}(t-\tau')\right) \\ &= D\,\alpha\, f_1(0)\left(\mathrm{e}^{-D\,\alpha\,(t-\tau)}\, \mathrm{H}(t-\tau)\right. \\ &\quad \left. -\mathrm{e}^{-D\,\alpha\,(t-\tau)}\, \mathrm{H}(t-\tau') + \mathrm{e}^{-D[\alpha\,(t-\tau) + \beta\,(t-\tau')]})\, \mathrm{H}(t-\tau')\right). \end{aligned}$$

We deduce, from Equation (67), that

$$f_2(t) = \begin{cases} f_2(0) & \text{if } t < \tau \\ \mathrm{C}_1^{\mathrm{te}} + f_1(0)(1 - \mathrm{e}^{-D\,\alpha\,(t-\tau)}) & \text{if } \tau \leq t < \tau' \\ \mathrm{C}_2^{\mathrm{te}} + \frac{\alpha}{\alpha+\beta}\, f_1(0)\left(\mathrm{e}^{-D\,\alpha\,(\tau'-\tau)} - \mathrm{e}^{-D[\alpha\,(t-\tau) + \beta\,(t-\tau')]}\right) & \text{if } \tau' \leq t \end{cases}$$

The continuity of the function f_2 at times τ and τ' allows to find an equation for C_1^{te} and one for C_2^{te}. We therefore derive the function f_2 given by

$$\begin{aligned} f_2(t) =& f_2(0) + f_1(0)\left(\left[1 - \mathrm{e}^{-D\,\alpha\,(t-\tau)}\right]\mathrm{H}(t-\tau)\right. \\ & \left.\frac{\alpha}{\alpha+\beta}\left(\mathrm{e}^{-D\,\alpha\,(\tau'-\tau)} - \mathrm{e}^{-D[\alpha\,(t-\tau)+\beta\,(t-\tau')]}\right)\mathrm{H}(t-\tau')\right). \end{aligned}$$

We now derive expression for f_3. We find that

$$f_3'(t) = D\,\beta\,f_1(0)\,\mathrm{e}^{-D[\alpha\,(t-\tau)+\beta\,(t-\tau')]}\,\mathrm{H}(t-\tau').$$

The resolution leads to

$$f_3(t) = f_3(0) + \frac{\beta}{\alpha+\beta}\,f_1(0)\left(\mathrm{e}^{-D\,\alpha\,(\tau'-\tau)} - \mathrm{e}^{-D[\alpha\,(t-\tau)+\beta\,(t-\tau')]}\right)\mathrm{H}(t-\tau').$$

It is interesting to note that the symmetry is broken in the long-run

$$\begin{aligned} \lim_{t\to+\infty} f_2(t) &= f_2(0) + f_1(0)\left(1 - \tfrac{\beta}{\alpha+\beta}\,\mathrm{e}^{-D\,\alpha\,(\tau'-\tau)}\right), \\ \lim_{t\to+\infty} f_3(t) &= f_3(0) + f_1(0)\,\tfrac{\beta}{\alpha+\beta}\,\mathrm{e}^{-D\,\alpha\,(\tau'-\tau)}, \end{aligned}$$

and, quite importantly, we can demonstrate that

$$f_1(t) + f_2(t) + f_3(t) = f_1(0) + f_2(0) + f_3(0), \quad \forall t \in \mathbb{R}_+.$$

The conservation of information is held at all time.

9.4. Delay in the Differential Equation—Practical Case

We will need the following results, which generalize Definition 12 (specifically Equation (52)) and Theorem 4.

9.4.1. The Time-Dependent Directed Diffusion Equation Involving the Operator $L(t)$

Definition 14. *Let $f : E \times \mathbb{R}_+ \to \mathbb{R}$ representing the information function for a graph with n nodes. If D is the diffusion coefficient and $\Gamma : E \times \mathbb{R}_+ \to \mathbb{R}$ the creation function of the information, with a time-dependent Laplacian operator $L(t)$, the following directed $diffusion$ equation holds, for any $t \in \mathbb{R}_+$:*

$$\frac{\partial f}{\partial t}(t,x) = -D\,{}^tL(t)\,f(t,x) + \Gamma(t,x) \tag{70}$$

Here the main difference is that the Laplacian operator L is a function of time t.

9.4.2. Solution Involving $L(t)$

From Equation (12), we can use the Cauchy theorem to obtain the following solution.

Theorem 9 (Application of the Cauchy Theorem)**.** *The solution of the related Cauchy problem associated with Equation (12) is given by*

$$f(t,x) = \mathrm{e}^{-D\,\int_0^t({}^tL(s)\mathrm{d}s)}\,f(0,x) + \int_0^t \mathrm{e}^{-D\,\int_\tau^t({}^tL(s)\mathrm{d}s)}\,\Gamma(\tau,x)\,\mathrm{d}\tau. \tag{71}$$

As was done previously, we would rather use the following equation:

$$f(t+\Delta t,x) = \mathrm{e}^{-D\,\int_t^{t+\Delta t}({}^tL(s)\mathrm{d}s)}\,f(t,x) + \int_t^{t+\Delta t} \mathrm{e}^{-D\,\int_\tau^t({}^tL(s)\mathrm{d}s)}\,\Gamma(\tau,x)\,\mathrm{d}\tau$$

which we can reduce again by noticing that $\int_t^{t+\Delta t}({}^tL(s)\mathrm{d}s) = {}^tL(t)\Delta t + \mathrm{o}(\Delta t)$ and $\int_\tau^t({}^tL(s)\mathrm{d}s) = {}^tL(\tau)\Delta t + \mathrm{o}(\Delta t)$ since $t - \tau \leq \Delta t$, so that Equation (64) is also verified:

$$f(t+\Delta t, x) = \mathrm{e}^{-D\,{}^tL(t)\,\Delta t} f(t,x) + \Gamma(t,x)\,\Delta t + \mathcal{E}(t,x). \tag{72}$$

9.4.3. Practical Case—Committed Error

Here, the error $\mathcal{E}(t,x)$ committed due to the approximation being split into two terms, and one has

$$\mathcal{E}(t,x) = \mathcal{E}_{\mathrm{D}}(t,x) + \mathcal{E}_{\Gamma}(t,x),$$

where

$$\mathcal{E}_{\mathrm{D}}(t,x) = \mathrm{e}^{-D\,\int_t^{t+\Delta t}({}^tL(s)\mathrm{d}s)} f(t,x) - \mathrm{e}^{-D\,{}^tL(t)\,\Delta t} f(t,x)$$

is the error due to the approximation at the diffusion term, and

$$\mathcal{E}_{\Gamma}(t,x) = \int_t^{t+\Delta t} \mathrm{e}^{-D\,\int_\tau^t({}^tL(s)\mathrm{d}s)}\,\Gamma(\tau,x)\,\mathrm{d}\tau - \Gamma(t,x)\,\Delta t$$

is the error due to the approximation at the creation/destruction term. We derive an upper bound for these two errors. Regarding the error on the diffusion term, we note that

$$\int_t^{t+\Delta t}({}^tL(s)\mathrm{d}s) = \sum_{n=1}^{+\infty} \frac{(\Delta t)^n}{n!}\,{}^tL^{(n-1)}(t),$$

so that

$$\begin{aligned}
\mathcal{E}_{\mathrm{D}}(t,x) &= \left[\exp\left(-D\sum_{n=1}^{+\infty}\frac{(\Delta t)^n}{n!}\,{}^tL^{(n-1)}(t)\right) - \exp\left(-D\,{}^tL(t)\,\Delta t\right)\right] f(t,x) \\
&= \left(1 - D\,\Delta t\,{}^tL(t) + \frac{D}{2}\,(\Delta t)^2\,{}^tL'(t) + \frac{D^2}{2}\,(\Delta t)^2\,({}^tL(t))^2\right) f(t,x) \\
&\quad - \left(1 - D\,\Delta t\,{}^tL(t) + \frac{D^2}{2}\,(\Delta t)^2\,({}^tL(t))^2\right) f(t,x) + \mathrm{o}((\Delta t)^2) \\
&= \frac{1}{2}\,(D\,\Delta t\,{}^tL(t))^2 f(t,x) + \mathrm{o}((\Delta t)^2).
\end{aligned}$$

This proves that $\mathcal{E}_{\mathrm{D}}(t,x) = \mathrm{o}(\Delta t)$. Regarding the error on the creation/destruction term, we introduce

$$\gamma(\tau) = \mathrm{e}^{-D\,\int_\tau^t({}^tL(s)\mathrm{d}s)}$$

so that

$$\mathcal{E}_{\Gamma}(t,x) = \int_t^{t+\Delta t} \gamma(\tau)\,\Gamma(\tau,x)\,\mathrm{d}\tau - \Gamma(t,x)\,\Delta t.$$

We have

$$\gamma'(\tau) = D\,{}^tL(\tau)\,\gamma(\tau).$$

Bearing all above in mind, we have

$$
\begin{aligned}
\mathcal{E}_\Gamma(t,x) &= \sum_{n=1}^{+\infty} \frac{(\Delta t)^n}{n!} \left(\gamma(t)\,\Gamma(t,x)\right)^{(n-1)} - \Gamma(t,x)\,\Delta t \\
&= \gamma(t)\,\Gamma(t,x)\,\Delta t + \frac{(\Delta t)^2}{2}\left(\gamma'(t)\,\Gamma(t,x) + \gamma(t)\Gamma'(t,x)\right) - \Gamma(t,x)\,\Delta t + \mathrm{o}((\Delta t)^2) \\
&= \frac{(\Delta t)^2}{2}\left(\gamma'(t)\,\Gamma(t,x) + \gamma(t)\Gamma'(t,x)\right) + \mathrm{o}((\Delta t)^2) \\
&= \frac{(\Delta t)^2}{2}\left(D^{\,t}L(t)\,\Gamma(t,x) + \Gamma'(t,x)\right) + \mathrm{o}((\Delta t)^2).
\end{aligned}
$$

This proves that $\mathcal{E}_\Gamma(t,x) = \mathrm{o}(\Delta t)$. Finally, we have proven that the committed error is such that $\mathcal{E}(t,x) = \mathrm{o}(\Delta t)$. Thus, Equation (72) can be implemented by neglecting $\mathcal{E}(t,x)$ as this represents a negligible error (see Section 8.3 for numbers).

9.4.4. Practical Case—Implementation

3-Node Case

We implement the model through Equation (72). We took a 3-node graphs as a first step, and wanted to plot the solutions for the case depicted in Section 9.3.3. We set $f_1(0) = 1$ and $f_2(0) = f_3(0) = 3$, with $D = 3$, and we set the adjacency matrix A as

$$
A = \begin{pmatrix} 0 & 0.4 & 0.6 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.
$$

Figure 14 shows the information evolution per node (yellow is node 1; orange is node 2; and red is node 3).

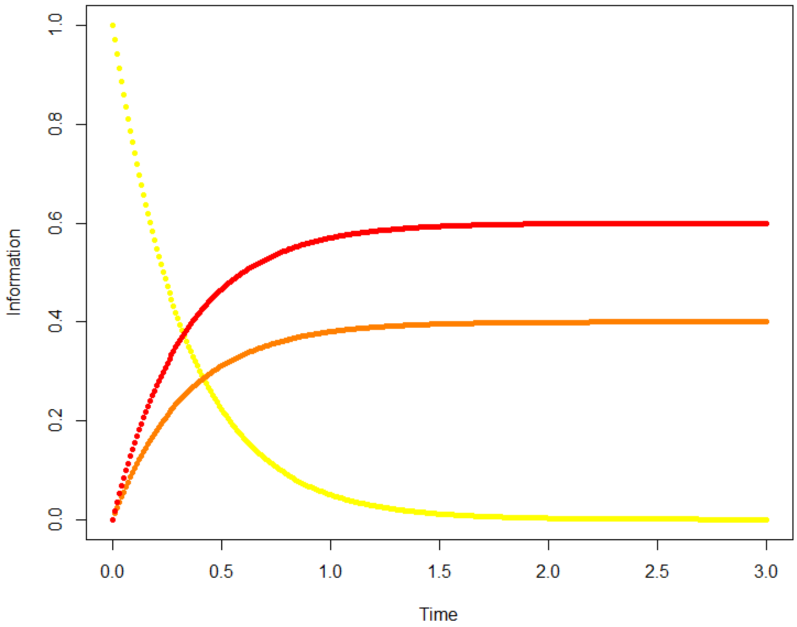

Figure 14. A 3 node-graph. Node 1 gives information to node 2 (resp. 3) with rate 0.4 (resp. 0.6). There is no delay, and $D = 3$.

We compare these results with the ones obtained from adding delays: node 2 starts receiving the information from time $\tau = 0.3$ and node 3 from time $\tau' = 0.9$. The time-dependent adjacency matrix becomes

$$A(t) = \begin{pmatrix} 0 & 0.4\,\mathrm{H}(t-0.3) & 0.6\,\mathrm{H}(t-0.9) \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

Figure 15 plots the results.

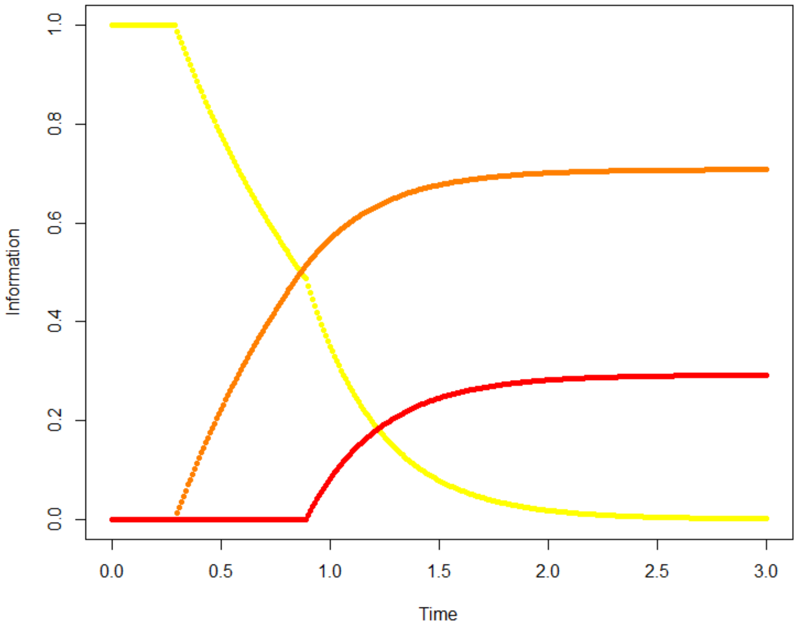

Figure 15. Same structure as in Figure 14, but with a delay $\tau = 0.3$ for node 2 and $\tau' = 0.9$ for node 3.

The results are showing expected behaviors, with another interesting aspect: if a node takes too much delay to receive information, even if it has strong affinity, it will receive much less than the expectations. This could modify the whole network configuration and change its dynamical properties.

A last remark is about the obvious discontinuity of the derivative at times where nodes start receiving the information.

Random Walk

Random walk is an important modelization of the propagation of information in a network. It represents the abrupt movement of a bunch of bitcoins (resp. epidemia) among different addresses (resp. people), and is a powerful tool for the traceability of sub-signals of one signal.

From what we have seen above, we can confidently state that random walk actually is a particular case of the above delayed differential equation theory. Indeed, a random walk is obtained by taking the limit $D \to +\infty$, with delay times as a multiple of a fixed delay $\tau > 0$, marking the time step of the movement of information from node to node (creating a discretization of time from a continuous time environment). It is worth pointing out that, in this case, diffusion becomes propagation. Thus, more generally, on a network, it turns out that a signal propagation is a particular diffusion of it, which is a non-trivial statement (but the converse is incorrect, as the sub-signals are not modified in essence through their propagation throughout the network).

10. Conclusions

In this paper, we studied the process of diffusion on a network, undirected and directed, with boundary conditions and response delays. More specifically, we introduced some fundamental definitions in the context of a crypto's P2P network, and then derived the diffusion equation, with two different Laplacian operators. The main innovations are the analytic solution derivation (by means of singular value decomposition), with or without boundary conditions. We also have characterized delays into the response of vertices.

It is worth pointing out that a tool, based on the evolution of information within a real blockchain network, could be envisaged through this approach. For instance, a visualization of any kind of exchange in the Bitcoin network could be implemented. The tool can trace any piece of the exchange, from the past to the future, using the forecasting ability, and this at any instant. The forecast, thus, can be updated by feeding the system with all historical information up to the present. We also can imagine a parametrized machine learning process which captures the historical configurations of the Bitcoin network, and suggests likely ones in the future. Thus, the traceability of information is a starting point for further transparency within all the agents, and implies the diminution of money-laundering risk. Modeling the P2P network is useful for any exchange, not only for trading and forecast purposes, but also for compliance duties.

Finally, it is worth stressing that the results developed in this paper are generalizable to any kind of P2P network, except Section 4.2, which is Bitcoin specific, even if the approach could be adapted. No matter what the underlying protocol is, we will systematically find exchange of information between agents, nodes, and users. The structure of the network as well as its underlying consensus mechanism (e.g., proof-of-work and proof-of-stake) are additional data which the general diffusion equation on a directed graph, given by Equation (52), can be a fitting model with parameter D, and, more importantly, with the operator Γ, function of the consensus, as Γ is related to the source of information. One could also make the model even more general by implementing a tensor D as a function of the vertices and of the edges of the network. This is left for further studies.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Notes

[1] We write $f : E \times \mathbb{R}_+ \to \mathbb{R}$. The function has two variables, i.e., time and node, and takes real values.

[2] numbers 1, 2, etc. designates the neighboring nodes of y.

References

An, Jaehyung, Alexey Mikhaylov, and Sang-Uk Jung. 2021. A Linear Programming approach for robust network revenue management in the airline industry. *Journal of Air Transport Management* 91: 101979. [CrossRef]

Bondy, J. Adrian, and Uppaluri S. R. Murty. 2007. *Graph Theory: An Advanced Course (Graduate Texts in Mathematics)*. Berlin and Heidelberg: Springer.

Chung, Soon-Yeong, Yun-Sung Chung, and Jong-Ho Kim. 2007. Diffusion and Elastic Equations on Networks. *Publications of the Research Institute for Mathematical Sciences* 43: 699–726. [CrossRef]

Hollanders, Romain, Daniel F. Bernardes, Bivas Mitra, Raphaël M. Jungers, Jean-Charles Delvenne, and Fabien Tarissan. 2014. Data-driven traffic and diffusion modeling in peer-to-peer networks: A real case study. *Network Science* 2: 341–66. [CrossRef]

Li, Quan-Lin, Jing-Yu Ma, Yan-Xia Chang, Fan-Qi Ma, and Hai-Bo Yu. 2019. Markov processes in blockchain systems. *Computational Social Networks* 6: 5. [CrossRef]

Lipton, Alexander, and Adrien Treccani. 2021. *Blockchain And Distributed Ledgers: Mathematics, Technology, And Economics*. Singapore: World Scientific.

Manini, Daniele, and Marco Gribaudo. 2008. An Analytical Study of the Resource Diffusion in Non-homogeneous P2P Networks. Paper presented at International Conference on Analytical and Stochastic Modeling Techniques and Applications, Nicosia, Cyprus, June 4–6. Berlin and Heidelberg: Springer, pp. 88–100.

Mikhaylov, Alexey Yu. 2021. Development of Friedrich von Hayek's theory of private money and economic implications for digital currencies. *Terra Economicus* 19: 53–62. [CrossRef]

Musa, Ahmad, Hamad Almohannadi, and Jassim Alhamar. 2018. Malware Propagation Modelling in Peer-to-Peer Networks: A Review. Paper presented at 2018 6th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW), Barcelona, Spain, August 6–8. pp. 198–202.

Pacreau, Grégoire, Edmond Lezmi, and Jiali Xu. 2021. Graph Neural Networks for Asset Management. *IEEE Transactions on Engineering Management* 1–18. Available online: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3976168 (accessed on 15 December 2021).

Shahsavari, Yahya, Kaiwen Zhang, and Chamseddine Talhi. 2017. A Theoretical Model for Block Propagation Analysis in Bitcoin Network. *IEEE Transactions on Engineering Management* 1–18. [CrossRef]

Thanou, Dorina, Xiaowen Dong, Daniel Kressner, and Pascal Frossard. 2017. Learning Heat Diffusion Graphs. *IEEE Transactions on Signal and Information Processing over Networks* 3: 484–99. [CrossRef]

Viriyasitavat, Wattana, and Danupol Hoonsopon. 2019. Blockchain characteristics and consensus in modern business processes. *Journal of Industrial Information Integration* 13: 32–39. [CrossRef]