Alma Mater Studiorum - Università di Bologna

DOTTORATO DI RICERCA IN FISICA

Ciclo XXIX

Settore Concorsuale di afferenza: 02/D1

Settore Scientifico disciplinare: FIS/07

MATHEMATICAL PHYSICS TECHNIQUES FOR OMICS DATA INTEGRATION

Presentata da: Matteo Bersanelli

Coordinatore Dottorato:

Prof. Gastone Castellani

Relatore:

Prof. Armando Bazzani

Correlatore:

Dott. Ettore Mosca

Esame finale anno 2017

Contents

Part III
Applications 50

Introduction

Nowadays different types of high-throughput technologies allow us to collect information on the molecular components of biological systems. Each of such technologies (e.g. nucleotide sequencing, DNA-chips and protein mass spectrometry) is designed to simultaneously collect a large set of molecular data of a specific kind: e.g. nucleotide sequences, gene expression and protein abundances. The biological information retrieved is measured at different "omic" levels. The word omic comes from the suffix common to most of the layers of molecular information retrievable (e.g. genomics, transcriptiomics, proteomics, metabolomics). In order to draw a more comprehensive view of biological processes, experimental data made on different layers have to be integrated and analyzed. The complexity of biological systems, the technological limits, the large number of biological variables and the relatively low number of biological samples make integrative analyses a challenging issue. Hence, the development of methods for the integrative analysis of multi-layer datasets is one of the most relevant problems computational scientists are addressing nowadays.

In the first part of the work we describe the issues arising from the analysis of omics and multi-omics datasets focusing ourselves on the mathematical aspects. Several omics data integration methods are presented and broadly divided into categories in order to get a first synthetic glance of the most representative and promising techniques used for the analysis of complex multi-level biological data. In the literature we noticed a growing interest around network-based methods. With the word network-based we mean approaches that use graphs for modeling and analyzing relationships among omic variables. Networks allow to model the intricate cellular molecular interactions and to use it as a framework for the integrated analysis of layers of biological information. In particular we found that algorithms that propagate molecular information on networks are being proposed in several applications and are often related to actual physical models. In order to set up a general physical mathematical framework to study the exchange of information in biological networks we considered the chemical master equation (CME). The CME is adapted to be descriptive of a stochastic process taking place on the network. We show that the macroscopic behavior of the network CME consists of a dynamical system from which it is possible to build up efficient algorithms and define new pipelines for the analysis and integration of omics.

In this work we propose two novel network-based methods with applications to both synthetic datasets and prostate ardenocarcinoma (PRAD) data. In both the applications the deleterious molecular information (e.g. somatic mutations) are mapped on the protein-protein interaction (PPI) interactome. In the first application we defined a novel methodology with the purpose of extracting differentially enriched modules (DEM) from the interactome that uses first a network diffusion algorithm to propagate the omic information on the network, and then defines the network smoothing index (NSI) and network resampling (NR) in order to extract the significantly connected network regions carrying most of the differential molecular information between two classes of samples (DEM). In the second application we study how the deleterious molecular information alters the normal information flow represented as a random walk on the network. Differently from the first application, the nodes of the network carrying deleterious information are modeled as exchanging information with an external node whose inner and outer connections to the existing network represent a perturbation of the biological network. The impact such perturbation is measured comparing the non-perturbed information flow in the network and the perturbed one that is characterized by the exchange of information with the external node. We define a critical threshold on the basis of the spectral properties of the existing network that characterizes a macroscopic shift in the information flow on the network. In such fashion it is possible to measure to which degree a distribution of altered molecular information on a given network deviates the normal trajectories of information flow.

Part I
Omics and multi-omics data

Chapter 1

Omics and multi-omics data integration methods

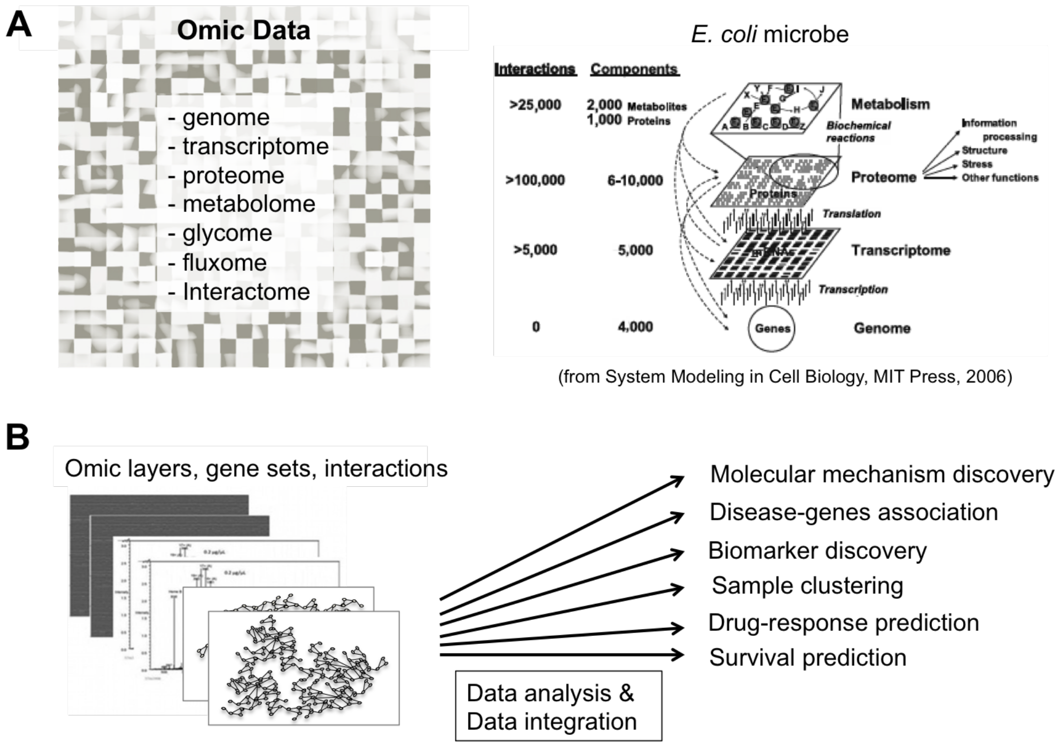
In this first chapter we describe the issues arising from the analysis of omics and multi-omics datasets focusing ourselves on the mathematical aspects. Several multi-omics data integration methods are presented and broadly divided into categories in order to get a first synthetic glance of the challenges arising from the analysis of complex multi-level biological data. Complete insights and further connections to other related literature can be found in the publeshed article *"Methods for the integration of multi-omics data, mathematical aspects"* [1].

1.1 Introduction to omics and multi-omics data

Biological functions are exploited by systems of interacting molecules and macromolecules that take part in physical and biochemical processes in structured environments. Different types of high-throughput technologies allow us to collect information on the molecular components of biological systems. Each of such technologies (e.g. nucleotide sequencing, DNA-chips and protein mass spectrometry) is designed to simultaneously collect a large set of molecular data of a specific kind: e.g. nucleotide sequences, gene expression and protein abundances. Therefore, in order to draw a more comprehensive view of biological processes, experimental data made on different layers have to be integrated and analyzed. The complexity of biological systems, the technological limits, the large number of biological variables and the relatively low number of biological samples make integrative analyses a challenging issue. Hence, the development of methods for the integrative analysis of multi-layer datasets is one of the most relevant problems computational scientists are addressing nowadays.

A few reviews exist on this topic. For example, Berger *et al.* [2] described integrative approaches in one of the sections of their review, which is also focused on tools for the analysis of single omics layers, while Kristensen *et al.* [3] presented objectives, methods and computational tools of integrative genomics, with a particular focus on the applications related to cancer research. Conversely, we would like to focus on mathematical aspects and illustrate the solutions found to the problem of multi-omics data integration.

The classification of the approaches presented in the literature as multi-omics methods is a non-trivial task for at least three reasons. First, most of the computational



(from System Modeling in Cell Biology, MIT Press, 2006)

Figure 1.1: **Overview of omics data. A.** Omic data are seen as complementary layers of molecular information. On the right the complexity and the order of magnitude of the retrievable data for *E. Coli* microbe as an example. **B.** Main objectives of omic data integration

approaches developed so far are pipelines of analysis that apply several methods to carry out a sequence of tasks; therefore, different pipelines share some methods: for example, partial least squares regression is included in both Integromics [4] and sMBPLS [5]. Second, pipelines presented for addressing a particular problem can be also used, with minor modifications, to solve another problem, possibly with other types of omics. Third, several tools can be used in a supervised or unsupervised setting, according to the formulation of the problem.

On the basis of methodological aspects, we will consider two main criteria. The first is whether the approach uses graphs to model the interactions among variables. These approaches, designated as “network-based” (NB), take into account currently known (e.g. protein-protein interactions) or predicted (e.g. from correlation analysis) relationships between biological variables. In this class, graph measures (e.g. degree, connectivity, centrality) and graph algorithms (e.g. sub-network identification) are used to identify valuable biological information. Importantly, networks are used in the modeling of the cell’s intricate wiring diagram and suggest possible mechanisms of action at the basis of healthy and pathological phenotypes [6].

The second criterion is whether the approach is bayesian (BY) [7], that is, it uses a statistical model in which, starting from an *a priori* reasonable assumption about the data probability distribution, *parametric* or *non-parametric*, it is possible to compute the updated posterior probability distribution making use of the Bayes’ rule; of course the posterior distribution depends on dataset measurements [8]. In the network-based area, bayesian networks [9, 10, 11] are another promising solution for the analysis multi-omics data.

Therefore, we will arrange integrative methods in four classes: network-free non-

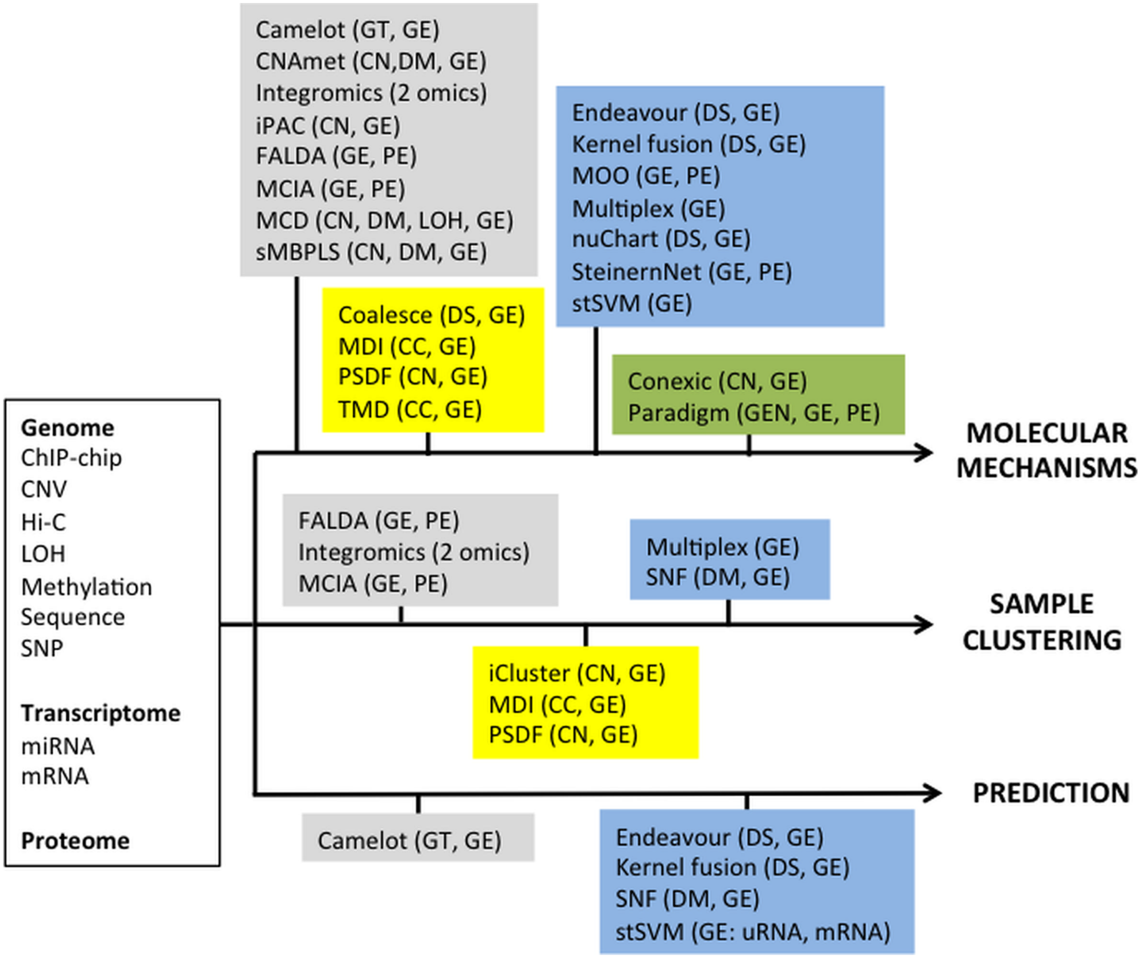


Figure 1.2: **Overview of omics and multi-omics methods.** Methods are placed in boxes according to whether they make use of networks and bayesian theory; the types of omics that each method takes in input (or has been applied to in a case study) is indicated between parentheses. Grey: network-free, non-bayesian methods; yellow: network-free, bayesian methods; blue: network-based, non-bayesian methods; green: network-based bayesian methods. Abbreviations: GEN = genome, CC = ChIP-chip, CN = copy number variations, DM = DNA methylation, DS = DNA sequence, Hi-C = genome-wide data of chromosomal interactions, LOH = loss of heterozigosity, GT = genotype, GE = gene expression, PE = protein expression.

bayesian (NF-NBY), network-free bayesian (NF-BY), network-based non-bayesian (NB-NBY) and network-based bayesian (NB-BY) methods. We will give an overview of the methods that have been proposed for the analysis of at least two different types of omics datasets and get an insight of the specific mathematical grounds. In particular, we choose to consider in detail the mathematical aspects of the most common, representative or promising methods of each category.

1.1.1 Methods overview

Mathematically, the general problem of analyzing multiple omics datasets can be formulated as the sequential or joint analysis of multiple component-by-sample matrices, possibly using other data that carry prior information on components and samples.

The objectives of integrative analysis can be summarized into the following [3] (Fig. 1.2): (i) the discovery of molecular mechanisms; (ii) the clustering of samples (e.g.

individuals); (iii) the prediction of an outcome, such as survival or efficacy of therapy. Most of the methods are developed for the first and second objectives, while less methods carry out prediction.

Integrative approaches can be more or less stringent on the types of omics considered in input: some methods are designed to analyze a specific combination of datasets, while others are more general. For example, Conexic [12] is tailored for DNA copy number variations (CNV) and gene expression data, while iCluster [13] can be in principle used for the analysis of any combination of omics encoded as quantitative values on the same set of samples (Tab. 1).

As already mentioned, a distinction can be done between *sequential* and *simultaneous* analysis of multiple layers. In the former case, the results of the analysis of one layer are refined by means of the subsequent analyses of further layers. This is the case, for example, of methods that are designed assuming a causal effect of an omics (e.g. genomics) on another (e.g. transcriptomics), like MCD [14] and iPAC [15]. The joint analysis of multiple omics can be carried out by means of models that consider each layer as a separate entity: two examples are multivariate regression [16] and multi-objective optimization [17, 18]. Simultaneous analysis may require a preliminary step of data fusion, which usually involves objects derived from single-layer analysis: two examples are the fusion of sample-sample similarity matrices [19] and of gene-gene kernels matrices [20] calculated on different omics.

1.2 Network-free non-bayesian (NF-NBY)

Among the approaches that have been developed for specific types of omics there are iPAC [15], MCD [14], CNAmet [21], sMB-PLS [5] and Camelot [16]. iPAC [15] is an unsupervised approach for the sequential analysis of CNV and gene expression data on the basis of a series of gene selection criteria: aberrant genes identified by the analysis of CNV are further studied by correlation analysis of gene expression in order to find the subset of aberrant genes potentially leading to a substantial shift in transcriptional programs. MCD [14] (Multiple Concerted Disruption) is another sequential approach. CNVs, loss of heterozygosity (LOH) and DNA methylations are analyzed sequentially in order to find changes in gene copy number accompanied by allelic imbalances and variations in DNA methylation resulting in gene expression differences. CNAmet [21] uses gene-wise weights calculated considering the gene expression in classes of samples with different CNVs or DNA methylation pattern; weights for CNV and DNA methylation are then linearly combined to define gene-wise statistics, whose significance is assessed by permutation analysis. In 2012 Li *et al.* presented the sparse Multi-Block Partial Least Squares (sMB-PLS) regression method [5] for the identification of regulatory modules from multiple omics. Common weights are found in order to maximize the covariance between summary vectors of the input matrices (CNV, DNA methylation and miRNA expression) and the summary vector of the output matrix (mRNA expression). A multi-dimensional regulatory module contains sets of regulatory factors from different layers that are likely to jointly contribute to a “gene expression factory”. Camelot [16] finds the optimal regression model for phenotype prediction (drug response) on the basis of matched genotype and gene expression data. This method suggests the molecular mechanisms that predict the phenotype under analysis.

Conversely from the methods above, Integromics [4], MCIA [22] and the approach of

Liu *et al.* [23] are based on models of data integration that can be easily applied to different types of omics. Integromics [4] performs integrative analysis of two types of omics with the main objective of finding similarities among samples and correlation among molecular components. It uses a regularized version of canonical correlation analysis to highlight correlations between the two datasets and a sparse version of partial least squares regression that includes simultaneous variable selection in both datasets. In principle, it can be applied to any pairs of omics that can be encoded as continuous sample-by-components matrices. Multiple co-inertia analysis MCIA [22] is an exploratory data analysis method that identifies co-relationships between multiple high-dimensional datasets. Based on a covariance optimization criterion, MCIA simultaneously projects several datasets into the same dimensional space, transforming diverse sets of features onto the same scale. This analysis leads to the identification of biological markers and clusters of samples. Liu *et al.* [23] presented a method (shortly FALDA) based on standardization and merger of several omics (namely mRNA, miRNA and protein data) into a joint (standardized) molecule-by-sample matrix. Then, factor analysis (FA) and linear discriminant analysis (LDA) are used to highlight molecular mechanisms that discriminate different classes of samples.

Many variations of PLS, a common dimensionality reduction method, have been introduced for the integration of complex datasets: for example, Integromics [4] relies on a sparse version of PLS (sPLS), and other variants of PLS, such as Orthogonal PLS [24], Kernel PLS [25] or O2-PLS [26], have been described in the literature. The idea of weighting the behavior of a gene at different levels and then combining such weights in order to get an integrated picture, applied so far for gene expression, CNV and methylation data [21], is a versatile approach that can be applicable to other types of datasets (e.g. gene expression, somatic mutations and protein expression). Thus, below we will describe in more detail Partial Least Squares (PLS) and the use of signal-to-noise statistics for the integrative analysis of multiple datasets [21].

1.2.1 Partial least squares

PLS and PCA (Principal Component Analysis) are techniques that seek to identify a small set of features that work as predictors of the response dataset. While PCA works in a purely unsupervised fashion, PLS makes use of the response in order to find appropriate linear combinations of the predictors that define a new set of features. In PLS the coefficients of the linear combination are chosen so that the highest weight is assigned to variables that are most strongly correlated to the response. In this sense we can say that PLS is a supervised alternative to PCA, for details see [27].

Multi-block PLS [5] is a method for performing PLS on a multi-layered dataset. Like any supervised PLS regression problem, sMBPLS's set up consists of n (e.g. $n = 3$) input layers X_1, X_2, X_3 and a response dataset Y, where observations are made on the same set of samples. The goal is to identify MDRMs (Multi dimensional regulatory modules) that are column subsets of the input datasets on the same samples that are strongly associated to the response. First each layer is represented as the first PLS predictor for $i = 1, 2, 3$, $(\mathbf{Z}_i = X_i \cdot \mathbf{w}_i)$ and the response Y is treated the same way $(\mathbf{U} = Y \cdot \mathbf{v})$, where $\mathbf{w}_i$, $\mathbf{v}$ are the loadings and $\mathbf{Z}_i$ and $\mathbf{U}$ are the summary vectors or latent variables of respectively the input and response datasets. Then sMBPLS defines $\mathbf{Z} = b_1\mathbf{Z}_1 + b_2\mathbf{Z}_2 + b_3\mathbf{Z}_3$ that is a summary vector of the three datasets. The weights b_i are supposed to account for the contribution of the i-th dataset to the total covariance. Mathematically the problem can be described as finding the optimal parameters so that the covariance between input and response (summarized in $\mathbf{Z}$ and $\mathbf{U}$) is optimized. The results improve substantially by introducing a constraint or a penalization to the objective function that needs to be optimized: sMBPLS uses a Lasso penalization - many different penalization choices are possible (for details see e.g. [27]). The effect of this penalization is often called sparsity, meaning that negligible coefficients tend to be drawn to zero. So the final function to be maximized can be expressed as

$$\Omega(\mathbf{Z}, \mathbf{U}, \mathbf{w}_i, \mathbf{v}, \mathbf{b}) = cov(\mathbf{Z}, \mathbf{U}) - \sum_{i=1}^{3} \mathbf{P}_{\lambda_i}(\mathbf{w}_i) - \mathbf{P}_{\lambda_4}(\mathbf{v}) \tag{1.1}$$

with the further restrictions that vectors $\mathbf{w}_i, \mathbf{v}, \mathbf{b}$ must have norm equal to 1; here P_{λ_i} are the Lasso penalizations. In order to estimate the optimal parameters in (1.1) Li *et al.* develop an ad hoc algorithm [5].

1.2.2 Gene-wise weights

Multi-omics gene-wise weights have been proposed to fuse three types of omics into a unique summary score for each gene [21]. These scores s_i are defined using gene expression, DNA methylation and CNV data:

$$s_i = (w_i^{me} + w_i^{cn}) \cdot \epsilon_i, \tag{1.2}$$

where w_i^{me} and w_i^{cn} are measures of the expression difference of the i-th gene between samples with high and low values of DNA methylation w_i^{me} and CNV w_i^{cn}, while ϵ_i is a normalization term. More precisely, layer-specific weights for each gene are calculated using the mean and standard deviation of gene expression

$$w_i = \frac{m_{i,1} - m_{i,0}}{\sigma_{i,1} + \sigma_{i,0}}, \tag{1.3}$$

where the suffixes 1 and 0 indicate, respectively, samples having high and low values of the other omics (DNA methylation or CNV). In summary, each variable is associated with the sum of a set of signal-to-noise scores, each of which is calculated considering the means and standard deviations of the variable using two subsets of samples of a given dataset (e.g. gene expression) defined on the basis of the values of the same variable in another layer (e.g. CNV or methylation).

1.3 Network-free bayesian (NF-BY)

Parametric or “strict” bayesian frameworks assume that the prior probability distribution follows a specific model dependent on one or more parameters. If the prior fits the data well parametric bayesian methods usually outperform non-parametric ones. On the other hand, if the initial guess for the prior is hard or even impossible to formalize, non-parametric or distribution-free methods are preferred [8]. It is important to remark that non-parametric or distribution-free methods are characterized by the fact that - unlike their parametric counterpart - the priors are not identifiable with a given family of probability distributions depending on one or more parameters, since this family would be too large, therefore introducing the need of an alternative definition of the priors in which - roughly speaking - the parameters themselves are supposed to be random. In this context, Antoniak [28] defined Mixtures of Dirichelet Processes (DPM) a useful set of priors for many non-parametric problems, that was taken as a starting point for many recent works aiming at the integration of multi-omics, such as TMD [29], MDI [30], PSFD [31], while, for example, iCluster [13] is a parametric method. The choice between parametric and non-parametric models is often not arbitrary, but it is driven by the type of data to be modeled.

iCluster [13] and MDI [30] have been developed with the main objective of sample clustering and can be applied to different types of omics. iCluster [13] takes as input two or more matrices and finds multi-omics clusters jointly estimating, by means of a prior-posterior bayesian structure, the clustering Z, which is modeled as a Gaussian latent variable having layer-specific weights and parameters. MDI (multiple dataset integration) [30] carries out the same objective (clustering) using a bayesian approach to jointly estimate the parameters of Dirichelet Process Mixture models. These models are applied to find clusters and relevant genes (features).

An approach closely related to MDI is Savage's Transcriptional Modules Discovery (TMD) [29] who also adopts a mixture modeling approach, using hierarchical Dirichelet process to perform integrative modeling of two datasets. Conversely to MDI, TMD aims at the identification of molecular mechanisms.

Patient-Specific Data Fusion (PSDF) [31] extends the TMD model for assessing the concordance of biological signals of samples in the two datasets taken into account (CNV and gene expression data). PSDF can be used to shed light on molecular mechanisms and cluster samples.

Coalesce [32] is a combinatorial algorithm specifically developed for the identification of regulatory modules from the analysis of gene expression and DNA sequence data. The multi-omics probability for a gene to be included into a module is calculated combining omics specific probabilities through the Bayes' rule.

Since iCluster was introduced, it is often being cited by subsequent works as an innovative reference approach for multi-omics clustering of samples, while, as already said, MDI shares a multi-layer analysis approach (based on Dirichelet Process Mixture models) with other recent methods. Hence, we will focus on iCluster and MDI in the following.

1.3.1 Bayesian latent variable models

In 2009, Shen *et al.* developed a joint variable model for integrative clustering, naming the resulting methodology iCluster [13] . Considering N datasets referred to the same group of samples, iCluster formulates sample clustering as a joint latent variable that needs to be simultaneously estimated from multiple genomic data types. The first step is to capture the similarities among genomic information in each data set, so that the within-cluster variance is minimized. This task is performed by an optimization through PCA of the classical K-means clustering algorithm, with the additional advantage of reducing the dimensionality of the data: if k is the number of clusters, the dimensionality n of the genomic data is basically reduced to the first k-1 principal directions. Second, the clustering scheme in each layer is represented as a Gaussian latent variable model with the Gaussian latent component Z capturing the dependencies across the data types. Dealing with N different omics measurements on the same p samples $X_1, X_2, ..., X_N$, each one of dimension $p \times n_i$ with usually $p << n_i$, the model can be written in the following fashion:

$$X_i \quad = \quad W_i \cdot Z + \epsilon_i \tag{1.4}$$

where the matrices W_i are the $p \times k - 1$ weight matrices and ϵ_i are the independent error terms. After taking a continuous parametrization Z^* of Z and assuming $Z^* \sim N(0, I)$ and $\epsilon = (\epsilon_1, ..., \epsilon_N) \sim N(0, cov(\epsilon))$, likelihood-based inference is obtained through the Expectation-Maximization (EM) algorithm [33]. iCluster requires the number of desired clusters k as input for the algorithm.

Recently, Kirk *et al.* [30] presented a bayesian method for the unsupervised integrative modeling of multiple datasets. MDI integrates information from a wide range of different datasets and data types simultaneously. In a general N-components mixture model, the probability density for the data $p(X)$ is modeled using Dirichelet-multinomial allocation mixture model,

$$p(X) = \sum_{k=1}^{N} w_k \cdot \pi(X|\theta_k) \tag{1.5}$$

where w_k are the mixture proportions, θ_k are the parameters associated to the k-th component and π is a parametric density. Component allocation variables and some additional parameters - conversely from the TMD model [29] - are introduced in order to capture the dependencies among these models and find clusters of genomic entities having the same behavior in different datasets. The modeling structure of the multi-layer dataset exploits the mathematical connection between mixture models and Dirichelet Processes, a non-trivial problem: for details see [34]. In this way is possible to construct a prior probability for each dataset where the probability distribution is parametrized by component allocation variables. Inference on such parameters is performed through Gibbs sampling. Finally, in order to identify groups that tend to cluster together in multiple datasets, it is natural to exploit the posterior probability as a metric in order to decide whether or not a connection among each couple of genes is strong enough across the dataset.

Both MDI and iCluster carry out simultaneous integrative clustering of multiple omics datasets. However, in contrast to MDI, iCluster seeks to find a single common clustering structure for all datasets.

1.4 Network-based non-bayesian (NB-NBY)

Methods that we have assigned to this category make either use of molecular interaction data or use networks defined from correlation analysis.

SteinerNet [35], the method proposed by Mosca *et al.* [17, 18], stSVM [36] and nuChart [37] share a common strategy: the analysis of a multi-weighted graph that carry multi-omics information. SteinerNet [35] is a method that identifies molecular sub-networks using omics datasets and a given molecular network. In order to reconstruct response pathways, SteinerNet finds a solution to the prize-collecting Steiner tree (PCST) problem, a minimum-weighted subtree that find an optimal network subject to weights assigned to vertexes and edges on the basis of input datasets. Similarly, multi-objective optimization (MOO) has been recently proposed for the extraction of sub-networks enriched in multi-omics information [17, 18]. Sub-networks are extracted on the basis of multiple criteria applied to a network that encodes several layers of biological information as vertex and edge weights. Also stSVM (smoothed t-statistic support vector machine) method [36] loads gene-wise statistics from multiple omics (miRNA and mRNA) on a molecular network known *a priori*. Then, a network diffusion method is used to smooth the statistics according to network topology. Significant genes are then used to train a classifier (a SVM) that predicts the type of sample (e.g. early versus late disease relapse). NuChart [37] is a method for the annotation and statistical analysis of a list of genes with information relying on Hi-C data (genome-wide data of chromosomal interactions [38]). NuChart identifies Hi-C fragments by means of DNA sequencing data and creates gene-centric neighborhood graphs on which other omics data (e.g. gene expression) are mapped and jointly analyzed.

ENDEAVOUR [39] calculates gene-wise statistics from heterogeneous genome-wide data sources (including molecular interactions) and ranks genes according to their similarity to known genes involved in the biological process under analysis. Single layer prioritizations are then integrated into a global ranking by means of order statistics. In 2007 De Bie *et al.* [20] proposed a kernel-based data fusion method for gene prioritization, which operates in the same setting of ENDEAVOUR. Kernels representing gene information in each layer are linearly combined in order to fuse the information and identify disease genes.

SNF (Similarity Network Fusion) [19] is a method that computes and fuses patient similarity networks obtained from each omics separately, in order to find disease subtypes and predict phenotypes. Conversely from the other methods of this section, SNF uses sample-sample networks obtained from correlation analysis. The key step of SNF is to iteratively and simultaneously update the global patient similarity matrix of each layer using a local K-nearest neighbours (KNN) approach combined with the global similarity matrices of the other layers. Fusion is then completed by averaging the similarity matrices once the iterative upgrading is performed.

Recently, a type of multi-partite network (multiplex) has been introduced as a novel theoretical framework for network-based multi-layer integrative analysis [40]. Multiplex networks are multi-layer systems of vertexes that can be linked in multiple interacting and co-evolving layers. This approach has been proposed for the analysis of gene expression data in brain [41] and cancer [42]. In the second example, a sample-sample duplex (two-layers network) has been generated based on correlation between gene expression profiles, revealing structural similarities and differences between two classes of samples. Thanks to their general formalism, in principle multiplex networks can be applied to the joint analysis of several types of omics (e.g. one type of omics for each layer), also for multi-level clustering purposes [43].

In the following subsections, we will discuss in more detail network diffusion, fusion of similarity networks and heterogeneous/multiplex networks. Methods that simulate the diffusion of information throughout a network are being increasingly used, since they allow to study how the information (e.g. differential expression, sequence variations) initially available in one or more network components (vertexes) affects other network regions [44]. SNF [19] is a diffusion-based strategy that can be easily extended to the analysis of a wide range of multi-omics data. Heterogeneous and multiplex networks are promising frameworks for innovative multi-omics data analysis.

1.4.1 Diffusion processes on networks

Network diffusion algorithms define a vector of scores σ associated with network vertexes on the basis of initial conditions $\mathbf{x}_0$ and network topology τ, usually represented by the adjacency matrix A or the Laplacian matrix L of the graph.
An application of such techniques is found in stSVM [36], where a p-step random walk kernel K is used in order to smooth the t-statistics $\mathbf{x}_0$, which assess the differential expression of genes. The kernel is defined as

$$K = (\alpha \cdot I - L')^p \tag{1.6}$$

where α is a constant, L' is the symmetrically normalized Laplacian matrix of the graph and p is the number of random walk steps. The smoothing of the t-statistic $\mathbf{x}$ is simply computed using the kernel K:

$$\mathbf{x} = \mathbf{x}_0^T \cdot K \tag{1.7}$$

In this case the influence of a node on the network is controlled by the parameter p. Basically, the information initially available in each vertex is distributed to its neighbors by means of the application of K. For a deeper insight of diffusion kernels see [45].
In other diffusion models, the network-based scores $\sigma = \sigma(X_0, \tau)$ are the steady state solution of a discrete or continuous diffusion process on the network that can have either a deterministic or a stochastic interpretation. An example of such a technique is the network propagation algorithm [57] exploited in the work of Hofree *et al.* [46]: after mapping a patient mutation profile onto a molecular network, network propagation is used to “smooth” the mutation signal across the network. Network propagation uses a process that simulates a random walk on a network with restarts according to the function:

$$\mathbf{x}(t) = \alpha A' \cdot \mathbf{x}(t) + (1-\alpha)\mathbf{x}_0, \tag{1.8}$$

where $\mathbf{x}_0$ is a vector representing some kind of genomic information about a patient (in this case mutation signal), A' is the symmetrically normalized adjacency matrix capturing correlations among genes, and $\alpha \in (0,1)$ controls how much information is retained in the nodes with respect to how much is not. For $t \to \infty$ for each patient, the discrete array $\mathbf{x}_0$ is smoothed into a real-valued array $\sigma = \mathbf{x}(\infty)$.
Network diffusion processes are often based on an actual physical model, having the benefit of exploiting physical quantities and concepts to drive the setting of the parameters. For example Vandin and Upfal [47] presented a computationally efficient strategy for the identification of sub-networks considering the hydrodynamic model introduced by Qi *et al.* [48]: fluid is pumped into the source node s at a constant rate, diffuses through the graph along the edges, and is lost from each node at a constant first-order rate until a steady-flow solution is reached.
The presence of random walks on a graph allows connections to many other physical models. For example, another interesting framework is represented by electric circuits [49], where the relation between the random walk of electrons on a circuit

and Kirkhoff laws is exploited. eQed is a recent application of the latter [50]. Recently Mirzaev and Gunawardena have collected and rigorously demonstrated some of the most important mathematical results in the context of information dynamics in a linear framework, also suggesting a possible stochastic interpretation of such diffusion processes on the network in the Chemical Master Equation formalism [51].

1.4.2 Fusion of similarity networks

An interesting strategy to perform simultaneous network-based integration of omics is the one at the basis of SNF [19]. A number N of different patient similarity networks with associated global similarity matrices $P_{i,0}$ are defined from N datasets. Let's assume $N = 2$ for the sake of clarity. Then, for each layer a KNN local similarity matrix S_i is introduced in order to retain only robust information. Subsequently, global similarity matrices are smoothed by two parallel interchanging diffusion processes that consist of the upgrading of the global similarity matrices with respect to the local similarity matrices of the other layer:

$$\begin{aligned} P_1(t+1) &= S_1 \cdot P_2(t) \cdot S_1^T \\ P_2(t+1) &= S_2 \cdot P_1(t) \cdot S_2^T \end{aligned} \tag{1.9}$$

having initial condition $P_i(0) = P_{i,0}$. After convergence, the fused similarity matrix is then defined as the average of P_1 and P_2 . The result is a similarity matrix that can be viewed as the weighted adjacency matrix of a network built by fusing the similarity networks associated with each layer [19].

1.4.3 Heterogeneous networks and Multiplex

In the context of multi-omics data analyses, multiple (k) layers can be represented by means of k networks. In this context, we can distinguish between two kinds of formalism: heterogeneous networks and multiplex networks.

Heterogeneous networks consider k different kinds of nodes, each type corresponding to a different layer of biological information. In this framework, intra-layer connections and inter-layer connections are formally treated in the same way, even if they can be weighed differently. The multi-layered information is therefore somehow squeezed on just one dimension and the properties of the resulting graph can be used to manipulate the data. For example, for $k = 2$ we can have vertexes of genes layer $g_1, g_2, \ldots, g_n$ and proteins layer $p_1, p_2, \ldots, p_m$. The Laplacian matrix of this heterogeneous network is a $(n+m) \times (n+m)$ matrix:

$$L_{gp} = \begin{bmatrix} L_g & B_{gp} \\ B_{pg} & L_p \end{bmatrix}, \tag{1.10}$$

where L_g and L_p are the Laplacian matrices of respectively gene and protein layers, while the matrices B_{gp} and B_{pg} contain the information about inter-layer connections; in the case the graph is undirected $B_{pg} = B_{gp}^T$. An example of application of heterogeneous network for modeling gene-phenotype networks was presented by Li and Patra [52].

Multiplex networks [40] are instead multi-partite networks in which each of the k layers models a different information about the same set of vertexes $v_1, v_2, \ldots, v_n$.

For example, let us consider two omics, represented as a two-layered multiplex composed of two sample × sample networks, where the edges of each network are placed in function of the sample-sample correlations found in the associated omics. Then, it is possible to analyze inter-layer correlations by means of multilnks, a quantity that summarizes the connectivity of each pair of samples across the layers. More precisely, a multilink is a k-dimensional binary array whose i-th component is set to 1 if the two samples are connected in the i-th layer and 0 otherwise. The formalism of multilink is the basis to define weighted measures and overlaps of the multiplex networks and other physical quantities, such as entropy, which introduces a theoretical framework to quantify and detect the information stored in complex networks [40, 42].

1.5 Network-based bayesian (NB-BY)

In this section we deal with methods that can be classified as both network-based and bayesian; these features select mainly those methods that are somehow related to bayesian networks (BNs). BNs are probabilistic models composed of a graph and a local probability model that can be either parametric or not. BNs represent an important area of machine learning theory and many applications of this topic are found in diverse fields. BNs can be thought as a combination of network theory and probability theory.

Within the BN framework an important method for multi-omics data integration is Paradigm [53]. Its goal is the definition of patient-specific pathway activities by means of probabilistic inference. Each biological entity (gene, protein, etc.) is modeled as a factor graph that can be defined to host a wide range of multi-omics information, and is associated with a prior probability of being activated in a given pathway.

Conexic, a bayesian network-based algorithm, has been introduced for the identification of driver mutations in cancer through the integration of gene expression and CNVs [12]. Conexic is based on a bayesian scoring function that evaluates how each candidate gene, or a combination of genes, predicts the behavior of a gene expression module across tumor samples. Networks, more precisely regression trees, are used to encode regulation programs.

Below, we will focus on the theoretical setup of the BN developed by Paradigm [53].

1.5.1 Paradigm: an application of bayesian networks

The goal of Paradigm is the definition of an entities × samples matrix called IPA (inferred pathway activity) where IPA$_{ij}$ reports a score that accounts for how likely the biological entity i is activated/null/deactivated in sample j.

The model is network-based since correlations between data points are modeled as factor graphs $\Phi = (\phi_1, ..., \phi_m)$ that are used for assigning a probability for the genomic entities or variables $\mathbf{X} = (X_1, ..., X_n)$:

$$P_\Phi(\mathbf{X}) = \frac{1}{Z} \cdot \prod_{j=1}^{m} \phi_j(\mathbf{X}_j) \tag{1.11}$$

where Z is a normalization constant accounting for all of the possible settings of the variables $\mathbf{X}$ and $\mathbf{X}_j$ is a set constituted by x_j and its "parents" $Pa(x_j)$ that are the nodes that have a link directed to x_j in the network. It is important to underline that the number of features m is much less than $2^n - 1$ (the number of possible edges in the graph): this "sparsity" facilitates integration. In this way it is possible to assign to each gene's x_i activity a first a prior probability distribution and then probability distribution consistent with the dataset measurements D:

$$P_{\Phi}(\mathbf{x}_i = a, D) \propto \prod_{j=1}^{m} \sum_{S \subset_{A_i(a) \cup D} X_j} \phi_j(S) \tag{1.12}$$

where Φ is the fully specified factor graph, $S \subset_{A_i(a) \cup D} X_j$ are all the possible configurations consistent with both the dataset measurements D and the fact that gene i is activated ($A_i(a)$ is the the singleton assignment set $\{\mathbf{x}_i = a\}$); the proportionality constant is the same as equation (1.11). The junction free inference algorithm and the belief propagation algorithm are used to infer the probabilities while EM algorithm [33] is used to learn the parameters. After inference log odds of the posterior probability distribution are used to measure the activity of each gene.

1.6 Discussion and conclusions

Methods for the analysis of multiple layers of biological information pave the way for a more comprehensive and deeper understanding of biological systems. Indeed, several authors were able to show that the integration of multi-dimensional datasets leads to better results from a statistical and a biological point of view than single layer analyses. For example, using MCD, Charj *et al.* [14] showed that the integration of DNA copy number, LOH, DNA methylation and gene expression data permits the identification of a higher number of DNA explained gene expression changes and a set of genes that would have been missed in standard single layer analysis; Liu *et al.* [23] reported an improvement in the identification of pathways and networks integrating miRNA, mRNA and proteins; Wang *et al.* [19] showed that their network fusion approach applied to gene expression and DNA methylation lead to clusters of patients (corresponding to cancer subtypes) with significantly different survival rates.

A better understanding of the algorithms underlying integrative approaches is important for their correct application and further development. Network-based approaches use graphs for modeling and analyzing relationships among variables and are one of the most important classes of multi-omics methods. These approaches take advantage of algorithms for graph analysis. In particular, algorithms that propagate information on networks are being proposed in several applications and are often related to actual physical models. Networks allow to model the intricate cell's wiring diagram and to use it as a framework for the integrated analysis of layers of biological information. However the incompleteness of experimentally detected molecular interactions is still a significant limit. Further, better tools of analysis are required, because assumptions like normality and variable independence are often not fulfilled [6]. Multi-layer network-based frameworks, such as heterogeneous and multiplex networks, allow the definition of novel tools for the integration of omics. For example, the already mentioned methods of network diffusion can be extended to such frameworks in order to get multi-omics propagation scores, and new clustering algorithms could be developed based on these multi-layer relationships.

Moreover, multiple omics data can be naturally embedded in a heterogeneous network framework, for example metabolomics and genomics data, considering genes that codify for enzymes as inter-layer links, and intra-layer relationship given by *a priori* biological knowledge (like protein-protein interaction network) or by network reconstruction based on metabolomics and transcriptomics data.

Another class of interesting approaches relies on Bayes' rule. Multilevel bayesian models (parametric or not) are facing the multi-omics challenge by building frameworks that facilitate a biologically appropriate formalism for the assumptions on the prior distribution (e.g. factor graphs, mixture models) and by programming non-trivial and efficient algorithms for parameter estimation. Assuming the bayesian framework is an interesting choice because it reduces the integration to the estimate of a smaller set of parameters, simultaneously suggesting a clear integration scheme. A limitation of such models is that for parametric methods the output strongly depends on how well the prior distribution assumption is able to capture the core information of the given dataset. Distribution-free approaches do not have such a problem but sometimes tend to lack in accuracy. In the network-based context the application of bayesian networks represents an interesting compromise between networks and probability theory. The bayesian framework is promising also regarding the issue of noise, because errors have the possibility to be formally taken into account from the beginning of the analysis.

Not surprisingly, genomics and transcriptomics are the two omics for which many and more established approaches of multi-layer analysis exist. However, the availability of methods that are not tailored for specific types of omics extends the applicability of integrative approaches also to omics that are still less covered by specific methods, such as proteomics, metabolomics or glycomics.

One of the main limitations of integrative approaches is related to dimensionality. In fact, if on one hand more layers correspond to a more complete picture of the biological system, on the other hand the dimensionality of the problem increases. However, *a priori* information on the relationships among the components of the biological system should help in reducing false discoveries.

Several methods are implemented using R [54], confirming the prominent role of this programming language in the analysis of biological data, and Matlab [55]. The availability of well-documented and user-friendly implementations is a crucial factor for the usability and spread of interesting methods. However, there are still several cases in which software packages are not provided.

Table 1.1: **Methods for the analysis of multi-omics datasets**. Specificity (S/G) indicates whether the method was designed for a specific combination of omics (specific) or not (general). Impl. stands for implementation. Legend: MWG = multi-weighted graph; FA = factor analysis; LDA = linear discriminant analysis; CCA = canonical correlation analysis; PLS = partial least squares; DMA = Dirichelet multinomial allocation

Method	**S/G**	**Multi-omics approach**	**Impl.**
Camelot [16]	specific	bivariate predictive regression model	NA
CNAmet [21]	specific	multi-omics gene-wise scores	R
FALDA [23]	general	FA + LDA of a joint matrix	NA
Integromics [4]	general	Regularized CCA, sparse PLS	R
iPAC [15]	specific	sequential	NA
MCD [14]	specific	sequential	NA
MCIA [22]	general	multiple co-inertia analysis	R
sMBPLS [5]	general	sparse Multi-Block PLS regression	Matlab
Coalesce [32]	specific	multi-omics probabilities	C++
iCluster [13]	general	joint Gaussian latent variable models	R
MDI [30]	general	DMA mixture models	Matlab
PSDF [31]	general	hierarchical DMA mixture models	Matlab
TMD [29]	general	hierarchical DMA mixture models	Matlab
Kernel Fusion [20]	general	integration of omics-specific kernels	Matlab
Endeavour [39]	general	omics-specific ranks and order statistics	webserver
MOO [17, 18]	general	sub-network extraction on MWG	R
Multiplex [40]	general	joint analysis of multi-layered networks	NA
NuChart [37]	specific	analysis of a MWG	R
SNF [19]	general	similarity network fusion	Matlab, R
SteinerNet [35]	specific	sub-network extraction on MWG	webserver
stSVM [36]	specific	MWG	R
Paradigm [53]	general	multi-omics bayesian factor graphs	C++
Conexic [12]	specific	sequential	Java

Part II
Methematical models

Chapter 2

The chemical master equation framework

In this chapter we propose the theoretical formalism of the chemical master equation (CME) as a valuable ground to build up new data-integration methods or give a physcal meaning to some of the existing ones. In the first secition we derive the CME following the Van Kampen [56], in the second section we adapt the CME to a network dynamics in the form of a random walk; finally in the last section we show the connection between the master equation and the network propagation algorithm [57], a known network-diffusion algorithm that has been already exploited for omics data manipulation [46]. Both the biological applications described in further chapters (Part III) are referable to the CME framework.

2.1 General formulation of the chemical master equation

The chemical master equation is a useful stochastic equation to model the mesoscopic evolution of - in general - any Markov stochastic process Y that has the Markov property. Most of the results presented in this section are extensively treated in Van Kampen work [56]; we now try to focus on the aspects found to be most applicable to the omics data analysis context.

2.1.1 Markov processes

A stochastic Markov process is characterized by the "loss of memory" property which states that the probability of the realization (y_n, t_n) of the state n conditioned on (y_{n-1}, t_{n-1}) is uniquely determined and not affected by any of the values at earlier times. Formally, considering any set of n succesive times $t_1 < t_2 < \cdots < t_n$

$$P(y_n, t_n | y_{n-1}, t_{n-1}; y_{n-2}, t_{n-2}; \cdots ; y_1, t_1) = P(y_n, t_n | y_{n-1}, t_{n-1}) \tag{2.1}$$

A direct consequence of the Markov property is that the process is fully determined by the initial probability distribution and transition probability

$$P(y_1, t_1), \qquad P(y_2, t_2 | y_1, t_1) \tag{2.2}$$

All the hierarchy of probability distributions can be derived directly from (2.2). For simplicity in this section we assume each random variable Y_n to be one-dimensional, but the formulation will be generalized in the next section.

Lemma Condition (2.1) together with the obvious assumption

$$P(y_2,t_2) = \int P(y_2,t_2|y_1,t_1)P(y_1,t_1)dy_1 \tag{2.3}$$

Is equivalent to the Chapman-Kolmogorov equation

$$P(y_3,t_3|y_1,t_1) = \int P(y_3,t_3|y_2,t_2)P(y_2,t_2|y_1,t_1)dy_2 \tag{2.4}$$

Proof

Assuming $t_1 < t_2 < t_3$, from the Markov property we get

$$\begin{aligned} P(y_3,t_3;y_2;t_2;y_1,t_1) &= P(y_3,t_3|y_2t_2;y_1,t_1)P(y_2t_2;y_1,t_1) \\ &= P(y_3,t_3|y_2t_2)P(y_2t_2|y_1,t_1)P(y_1,t_1) \end{aligned}$$

Integrating over y_2 we get

$$\begin{aligned} P(y_3,t_3;y_1,t_1) &= P(y_1,t_1)\int P(y_3,t_3|y_2t_2)P(y_2t_2|y_1,t_1)dy_2 \\ P(y_3,t_3|y_1,t_1)P(y_1,t_1) &= P(y_1,t_1)\int P(y_3,t_3|y_2t_2)P(y_2t_2|y_1,t_1)dy_2 \end{aligned}$$

dividing both sides by $P(y_1,t_1)$ we get equation (2.4). The vice-versa is trivial □.

The Chapman-Kolmogorov equation therefore characterizes Markov processes.

2.1.2 Markov chains

The most simple but very useful class of Markov processes are the Markov chains. Formally, given a stochastic process Y they are characterized by the following properties:

i. Y has a discrete range.

ii. Y has a discrete time variable that take only integer values.

iii. Y is stationary or at least homogeneus.

We call a Markov chain *finite* when its range consists of a finite set of states $1,\cdots,M$. Finite Markov chains have been studied deeply in since they are the simplest Markov process that still presents most of the significant features. Considering now on only finite Markov chains, propertiy (iii.) reads

$$T_\tau = T^\tau, \qquad \tau = 0,1,2,\cdots \tag{2.5}$$

where the transition probability $T = T(y_2|y_1)$ is a M×M matrix. Given any initial probability distribution for the states $\vec{p}(0)$ (an array of M components), any probability distribution generating from $\vec{p}(0)$ can be written as

$$\vec{p}(t) = T^t\vec{p}(0) \tag{2.6}$$

Here $p_i(t)$ is the probability to find the system in state i at time t with $i = 1, 2, \cdots, M$. It is therefore clear that the study of finite Markov chains mostly reduces to explore the powers of the transition matrix T that in general is a stochastic matrix meaning that its entries are all non-negative elements and each column sums up to 1 because of the probabilistic interpretation of the variable $\vec{p}$. Perron and Frobenius [56] showed mathematically that, apart from some exceptional cases, that given any stochastic matrix T the process converges to a unique stationary distribution as t approaches ∞. These results will be exploited in next section for the random walk modelling.

2.1.3 Derivation of the chemical master equation

Let's consider a homogeneus Markov process Y; we can therefore use the notation T_τ for the transition probability that depends only on the time difference $\tau := t_2 - t_1$. We now derive the Master equation that is more appealing and physically interpretable than the Chapman-Kolmogorov equation (2.4). We assume the Markov process to be stationary (or at least homogeneus) so that we can simplify the notation $(P(y_2, t_2|y_1, t_1) = T_\tau(y_2|y_1))$, since the transition probability depends on the time difference alone $\tau = t_2 - t_1$. Equation (2.4) reads

$$T_{\tau+\tau'}(y_3|y_1) = \int T_{\tau'}(y_3|y_2)T_\tau(y_2|y_1)dy_2 \tag{2.7}$$

The Master equation basically consist of the continuous limit of the differential version of equation (2.4) as the time difference τ' vanishes. We assume that, for small values of $\tau' = t_3 - t_2$ the Taylor expansion of the transition probability takes the following form for small τ'

$$T_{\tau'}(y_3|y_2) = \delta(y_3 - y_2) + \tau' W(y_3|y_2) + o(\tau') \tag{2.8}$$

where $W(y_3|y_2)$ is the time derivative of the transition probability evaluated in $\tau' = 0$, therefore called the transition probability per unit time. In order to satisfy the normalizing condition $(\int T_{\tau'}(y_3|y_2)dy_3 = 1)$ we introduce a correction

$$T_{\tau'}(y_3|y_2) = (1 - a_0(y_2)\tau')\delta(y_3 - y_2) + \tau' W(y_2|y_1) + o(\tau') \tag{2.9}$$

where in the first right-hand side term the coefficient a_0 is

$$a_0(y_2) = \int W(y_3|y_2)dy_3 \tag{2.10}$$

so that $1 - a_0\tau'$ in front of the delta function represents the probability that no exchanges take place during the time interval τ'. We now substitute expression

(2.9) into (2.7) and get, exploiting the linearity of the integral operator

$$\begin{aligned} T_{\tau+\tau'}(y_3|y_1) &= \int T_\tau(y_2 - y_1)\delta(y_3 - y_2)dy_2 \\ &+ \tau' \int T_\tau(y_2|y_1)a_0\delta(y_3 - y_2)dy_2 \\ &+ \tau' \int T_\tau(y_2|y_1)W(y_3|y_2)dy_2 \end{aligned}$$

The second term on the right-hand side due to the definition (2.10) becomes $-\tau' \int W(y_2|y_3)T_\tau(y_3|y_1)dy$ while the first term on the right hand side becomes $T_\tau(y_3|y_1)$ after integration; we bring it to the left, divide both sides by τ' and take the limit $\tau' \to 0$ in order to get the Master equation:

$$\frac{\partial T_\tau(y_3|y_1)}{\partial \tau} = \int \left[W(y_3|y_2)T_\tau(y_2|y_1) - W(y_2|y_3)T_\tau(y_3|y_1)\right] dy_2 \tag{2.11}$$

that, omitting the discrete indices becomes

$$\frac{\partial P(y,t)}{\partial t} = \int \left[W(y|y')P(y',t) - W(y'|y)P(y,t)\right] dy' \tag{2.12}$$

When the range of Y is discrete (let's suppose M different states), the master equation reads:

$$\frac{\partial p_i(t)}{\partial t} = \sum_{j=1}^{M} \left[W_{ij}p_j(t) - W_{ji}p_i(t)\right] \tag{2.13}$$

where $p_i(t)$ is the probability to find the system in state i at time t. From equations (2.11, 2.12, 2.13) it is clear that the Master equation is a gain-loss equation for the probability for the system to be in a certain state (y or i) describing its time evolution.

2.1.4 The stationary distribution and detailed balance

One of the most important goals once the Master equation is derived is to find the stationary distribution that consists of the relaxation of the solution of equation (2.12, 2.13) for t going to ∞. We now refer only to equation (2.13) for simplicity. Many different proofs show how, in non-degenerate cases and when the space state is finite, for equation (2.13) it exist a unique stationary distribution. Some exceptions exist when we deal with an infinite number of states such as the random walk [56]. We do not want to treat this argument into detail; we just mention a few possibilities: a formal way to prove the existence and the uniqueness of the stationary distribution is using the time-discretiztion and then re-adapt the Perron-Frobenious theorem valid for Markov chains; a more "physical" approach is through an entropy function or a Lyapunov function optimization. Other proofs are due to Uhlmann and Kirchoff, who used network theory, while the Van Kampen itself gives a synthetic proof [56]. However in the formulation of the master equation for a random walk on a network the existence and uniqueness of the stationary distribution will follow directly from the Laplacian condition defined in the next section.

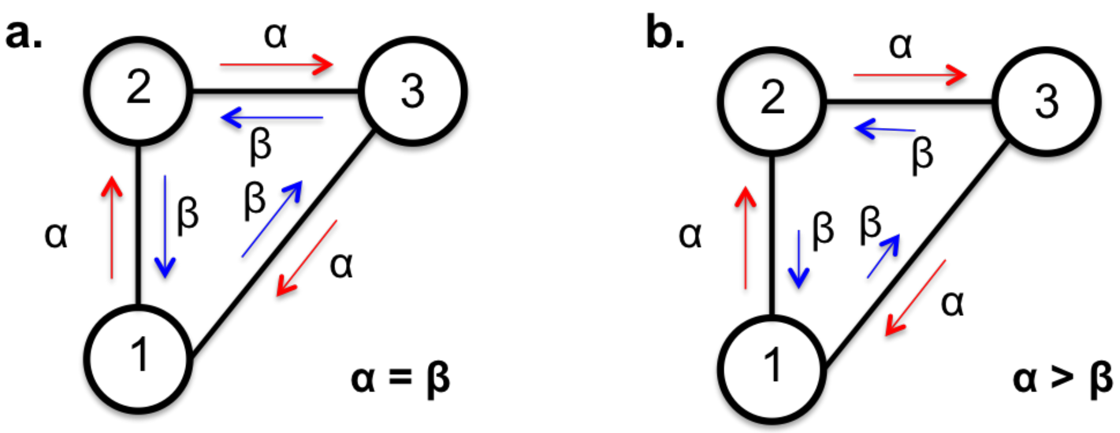


Figure 2.1: **Detailed Balance a.** System that satisfies the detailed balance condition **b.** System at stationary state not satisfying the detailed balance condition. α and β are the transition rates between states.

From equation (2.12) we see that that for the stationary distribution it must hold

$$\sum_j W_{ij} p_j(t) = \sum_j W_{ji} p_i(t) \tag{2.14}$$

which reflects the obvious fact that in the stationary condition the sum of all transitions to the state n from the other states n' and the transitions from n to other states n' must balance. The detailed balance condition asserts a stronger fact

$$W_{ij} p_j(t) = W_{ji} p_i(t) \quad \forall i, j \tag{2.15}$$

imposing that the transitions between any two states separately must balance. A simple example is shown in Fig. 2.1. Detailed balance condition (2.15) implies the stationary condition (2.14). We also remark that in quantum mechanics detailed balance follows from thermodynamic equilibrium. Therefore detailed balance is a necessary condition for thermodynamic equilibrium. In Van Kampen is proved that in closed isolated physical systems the stationary distribution of the master equation must satisfy the delailed balance condition. With the word "physical" we mean the system can be described microscopically in terms of Hamilton or Schrodinger equations. Closed and isolated respectively means that there's no exchange of particles with the environment and that no external force or field is acting on the system so that we can think of the energy of the system as a constant.

2.1.5 One-step processes

A one-step process is a Markov process in which transition probabilities assume a particularly simple and interpretable form. Such processes are also called generation-recombination or bith and death processes. We define a one-step process a Markov process with continuous time and discrete range whose transition matrix allows with high probability jumps only to adjacent states (the probability to jump two steps in a time interval Δt is $o(\Delta t)$), therefore giving rise to a tridiagonal Laplacian matrix where the off-diagonal transitions are given by:

$$W_{ij} = r_j \delta_{i,j-1} + g_j \delta_{i,j+1} \quad \text{if} i \neq j \tag{2.16}$$

while and on the diagonal we have

$$W_{ii} = -(r_i + g_i) \tag{2.17}$$

where r_i and g_i are the probabilities per unit time for a particle to jump respectively to the state $i-1$ and $i+1$ so that the general form of the master equation can be written as

$$\dot{p}_i = r_{i+1}p_{i+1} + g_{i-1}p_{i-1} - (r_i + g_i)p_i \tag{2.18}$$

that, using the Van Kampen step operators

$$E^{\pm} f(i) = f(i \pm 1), \quad \text{for any suitable function} f$$

becomes

$$\dot{p}_i = (E^+ - 1)r_i p_i + (E^- - 1)g_i p_i \tag{2.19}$$

The one step processes are classified depending on the range ($\mathbb{Z}$, $\mathbb{N}$ or finite $i = 0, 1, \cdots M$). In the last two cases appropriate boundary conditions are needed. Another classification arises looking at the coefficients r_n and g_n of (2.19):

i. *Random Walk* when both coefficients are constant.

ii. *Linear* One-step process when at least one between the coefficients is a linear functions of the state n.

iii. *Non-linear* One-step process when at list one coefficient is a non-linear function of the state n.

Remark The terms "linear" or "non-linear" here are always referred to the coefficients and not to the unknown p_i: the master equation is always a linear function of p_i.
An interesting property of One step processes (2.19) is that the stationary distribution assumes a particularly manageable form. For instance

$$p_i^s = \frac{g_{i-1}g_{i-2}\cdots g_1 g_0}{r_i r_{i-1}\cdots r_2 r_1} p_0^s \tag{2.20}$$

with the constraint

$$\frac{1}{p_0^s} = 1 + \sum_{i=1}^{M} \frac{g_{i-1}g_{i-2}\cdots g_1 g_0}{r_i r_{i-1}\cdots r_2 r_1} \tag{2.21}$$

Proof From equation 2.19 we can see that in the stationary solution it holds

$$\begin{aligned} (E^+ - 1)r_i p_i^s + (E^- - 1)g_i p_i^s &= 0 \\ (E^+ - 1)[r_i p_i^s - E^- g_i p_i^s] &= 0 \end{aligned}$$

where the quantity in the square brachets represents the probability flow J from i to $i-1$. Substituting into J the boundary conditions

$$\dot{p}_0 = r_1 p_1 - g_0 p_0$$

or

$$r_0 = g_{-1} = 0$$

we find $J = 0$ and therefore

$$r_i p_i^s = g_{i-1} p_{i-1}^s \tag{2.22}$$

so that by applying this relation iteratively one finds the stationary distribution (2.20) on all the range (starting from 0 and finishing with M). Such results can be easily extendend to both the half-infinite and two-sided infinite range. For the detailed proof of statements see Van Kampen. □

Remark In a close isolated physical system the detailed balance condition for one-step processes reads

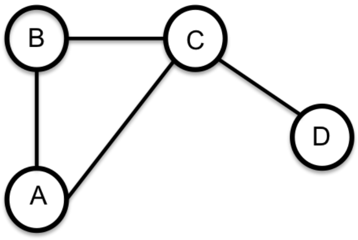
$$r_i p_i^e = g_{i-1} p_{i-1}^e \tag{2.23}$$

that has the same form as (2.22), implying that in such conditions the stationary solution $\vec{p}_s$ is equivalent to thermodynamic equilibrium $\vec{p}_e$. However we underline that this equivalence holds only for closed isolated systems, while condition (2.22) applies to open systems as well.

One-step processes of fundamental importance are the Poisson process, the decay process,the birth and death process and many other. We do not analize these Markov processes into detail, since it deviates from the main purposes of this work. However the applicability of one-step processes in the biological context would deserve more discussion; just to mention an example, the relative species abundance in rainforests, coral reefs [58] or the relative molecular species abundance in the gut microbiota [59] can be modelled as one-step processes. The major limitation for the applicability of one-step processes to omics data manipulation lies on the fact that the data must somehow fit in a sub-sequent set of states, which is not trivial given the complexity of the data and a network structure. In fact a one-step process is strictly one-dimensional: on a given network it is possible to define a one-step process only in restricted situations.

2.2 Random walk on networks

In the perspective of applying the Master equation mathematical framework to the context of omics and multi-omics data analysis, we now adapt the formalism described in the previous section to a physical situation in which molecular species that have state $\vec{n} = (n_1, \cdots, n_M)^T$ are forced to move and interact on a graph; the vertices represent the molecular species and the edges of the graph represent their (directed or undirected) interactions. In the physical context the word ”network” is generally preferred instead of the word ”graph”: network theory is a specific subject that has been applied to the study of Complex Systems. In the most general case the transition probabilities from a node i to another node j can be any (linear or non-linear) function of the number of molecular species n_i leading to a reaction-diffusion network that is usually hard to solve analytically [60]. Leaving such genaralizations and its implications to next sections we now focus on the random walk since it'll turn out to be at the basis of some important network-based algorithms exploited in the analysis of omic datasets. In this section we will develop a network-based version of the Master equation following as exploring guideline the arguments treated in the previous chapter.



Adjacency matrix

$$A = \begin{array}{c|cccc} & [A] & [B] & [C] & [D] \\ \hline {[A]} & 0 & 1 & 1 & 0 \\ {[B]} & 1 & 0 & 1 & 0 \\ {[C]} & 1 & 1 & 0 & 1 \\ {[D]} & 0 & 0 & 1 & 0 \end{array}$$

Degree matrix

$$G = \begin{array}{c|cccc} & [A] & [B] & [C] & [D] \\ \hline {[A]} & 2 & 0 & 0 & 0 \\ {[B]} & 0 & 2 & 0 & 0 \\ {[C]} & 0 & 0 & 3 & 0 \\ {[D]} & 0 & 0 & 0 & 1 \end{array}$$

Definitions:

1) $L_g = G - A$

2) $\pi = A\, G^{-1}$

3) $L = L_g G^{-1} = I - \pi$

4) $L' = G^{-1/2} L_g G^{-1/2}$
$= G^{-1/2} L\, G^{1/2}$

Graph Laplacian matrix

$$L_g = \begin{array}{c|cccc} & [A] & [B] & [C] & [D] \\ \hline {[A]} & 2 & -1 & -1 & 0 \\ {[B]} & -1 & 2 & -1 & 0 \\ {[C]} & -1 & -1 & 3 & -1 \\ {[D]} & 0 & 0 & -1 & 1 \end{array}$$

Laplacian matrix

$$L = \begin{array}{c|cccc} & [A] & [B] & [C] & [D] \\ \hline {[A]} & 1 & -1/2 & -1/3 & 0 \\ {[B]} & -1/2 & 1 & -1/3 & 0 \\ {[C]} & -1/2 & -1/2 & 1 & -1 \\ {[D]} & 0 & 0 & -1/3 & 1 \end{array}$$

Stochastic transition matrix

$$\pi = \begin{array}{c|cccc} & [A] & [B] & [C] & [D] \\ \hline {[A]} & 0 & 1/2 & 1/3 & 0 \\ {[B]} & 1/2 & 0 & 1/3 & 0 \\ {[C]} & 1/2 & 1/2 & 0 & 1 \\ {[D]} & 0 & 0 & 1/3 & 0 \end{array}$$

Normalized Laplacian matrix

$$L' = \begin{array}{c|cccc} & [A] & [B] & [C] & [D] \\ \hline {[A]} & 1 & -1/2 & -1/\sqrt{6} & 0 \\ {[B]} & -1/2 & 1 & -1/\sqrt{6} & 0 \\ {[C]} & -1/\sqrt{6} & -1/\sqrt{6} & 1 & -1/\sqrt{3} \\ {[D]} & 0 & 0 & -1/\sqrt{3} & 1 \end{array}$$

Figure 2.2: **Laplacian Matrices** With a simple toy network we illustrate the definitions choosen for this work.

For sake of clarity we illustrate the main matrices associated with a given undirected network with a simple example in Fig. 2.2. We remark that while the adjacency matrix, the degree matrix, the graph Laplacian are univocally defined, the other matrices are not unique (e.g. there are infinite stochastic matrices describing possible transitions on a given graph). We clarify the definitions since in literature the word "Laplacian" is used for many different concepts. In particular in the present work a Laplacian matrix is equivalent to the Van Kampen's $\mathbb{W}$ matrix [56].

2.2.1 Network master equation for the random walk

Considering an M nodes network, we introduce an $M \times M$ stochastic connection matrix π. Its entries π_{ij} describe the probability that in a given time interval Δt a particle jumps from node j to node i following an existing edge of the network. We consider the random walk of N particles, where the state of the system is summarized by the vector $\vec{n}$ with the constraint that the number of molecular species id fixed $N = \sum_i n_i(t) \quad \forall t$; it is therefore not considered the possibility of birth/death or creation/annihilation of the molecular species. In the general case the particles may interact with each other leading to non-trivial behaviors; if the particles do not interact the entries of the stochastic matrix π_{ij} can be seen as the probability distribution of independent random variables.

Let's now consider the single particle's a dynamic; we define $\delta_i^a(t)$, $a = 1, \cdots, N$

and $i = 1 \cdots , M$ so that

$$\delta_i^a(t) = \begin{cases} 1 & \text{if } a \text{ is in node } i \text{ at time } t \\ 0 & \text{otherwise} \end{cases}$$

and we can write respectively the probability to find a single particle in a given node i at time t and the number of species living on node i at time t as

$$p_i^a(t) = \langle \delta_i^a(t) \rangle, \qquad n_i(t) = \sum_a \delta_i^a(t) \tag{2.24}$$

here we average $\delta_i^a(t)$ over all the possible realizations of the process. We can now introduce the stochastic jump matrix Ψ^a that realizes the jump of the particle a along the existing network edges. It must hold that the average values of the jump matrix entries ψ_{ij}^a are the correspondent etries π_{ij} of the stochastic matrix. Formally Ψ^a needs to be a change of base in $\mathbb{R}^M$: a single particle state $\vec{\delta}^a(t) = (\delta_1^a(t), \cdots , \delta_M^a(t))^T$ is equal to $\vec{e}_i$ if and only if $\delta_i^a(t) = 1$. Given a time interval Δt the particle a is in node i at time $t + \Delta t$ only if

$$\delta_i^a(t + \Delta t) = \sum_j \psi_{ij}^a(t)\delta_j^a(t) \tag{2.25}$$

When we consider the N particles case, we naturally generalize the single particle jump matrix to the matrix Ψ

$$\psi_{ij} = \frac{1}{n_j(t)} \sum_a \psi_{ij}^a(t)\delta_j^a(t) \tag{2.26}$$

where $n_j(t) \neq 0$; here the normalization by $n_j(t)$ makes sure that the matrix Ψ is stochastic. The evolution of the state of node i, assuming that particles do not interact is therefore

$$n_i(t + \Delta t) = \sum_a \delta_i^a(t + \Delta t) = \sum_a \sum_j \psi_{ij}^a(t)\delta_j^a(t) = \sum_{j,n_j \neq 0} \psi_{ij}(t)n_j(t) \tag{2.27}$$

where, given equation (2.26) we need to assume that $n_j(t) \neq 0$ for any time value t. This is an intrinsic limit of the model: the number of particles N must be sufficiently big to leave a negligible probability to find a node completely empty. We will show later on that such probability decreases exponentially with N. Equation (2.27), since the columns of Ψ sum up to 1, can be written as:

$$n_i(t + \Delta t) - n_i(t) = \sum_{j,n_j \neq 0} \psi_{ij}(t)n_j(t) - \psi_{ji}n_i(t) \tag{2.28}$$

When we get the continuous limit of the evolution dynamics of the model we cannot assume complete independence among the particles, because of the possibility that in equations (2.26, 2.27, 2.28) to find $n_j(t) = 0$ for a certain time t.

In order to simplify the calculation but not eliminating the "empty node" issue just stated we assume a regularization condition. We set the probability for a couple of particles to jump simultaneousy in a time interval Δt to be proportional to Δt^2; formally

$$\begin{aligned}\pi_{ij} &= \hat{\pi}_{ij}\Delta t + o(\Delta t) \qquad i \neq j \\ \pi_{ii} &= 1 - \hat{\pi}_{ii}\Delta t = 1 - \Delta t \sum_{i \neq j} \hat{\pi}_{ij}\end{aligned} \tag{2.29}$$

so that, when we take the continuous limit only the contribution of the first order matters. Taking the average and having $\Delta t \to 0$ we get the mean-field equation:

$$\frac{d\langle n_i \rangle}{dt} = \sum_j \hat{\pi}_{ij}\langle n_j \rangle - \sum_j \hat{\pi}_{ji}\langle n_i \rangle \tag{2.30}$$

where the terms in the right-hand side are respectively the ingoing and outgoing flow of node i. Equation (2.30) with the regularization condition, limits the possible movements in the time interval Δt. This implies that any admissible combination of elementary exchanges $\Delta\vec{n} = \vec{n}(t + \Delta t) - \vec{n}(t)$ of independent particles has a probability distribution $p(\vec{n}, t)$ that can be derived by the Kolmogorov equation:

$$p(\vec{n}, t + \Delta t) = \sum_{\Delta\vec{n}} \pi(\Delta\vec{n}|\vec{n} - \Delta\vec{n})p(\vec{n} - \Delta\vec{n}, t) \tag{2.31}$$

where the sum is made on all the admissible combinations of elementary exchanges $\Delta\vec{n}$ leading to the state $\vec{n}$. In the regularization condition and assuming $\Delta t \to 0$ only one elementary exchange is possible therefore reducing $\Delta\vec{n} = \vec{e}_i - \vec{e}_j$ to the passage of a single particle from node j to node i so that

$$\pi(\vec{e}_i - \vec{e}_j|\vec{n}; \Delta t) = \frac{1}{N}\hat{\pi}_{ij}n_j\Delta t + o(\Delta t)$$

substituting this quantiy in equation (2.31) we obtain

$$p(\vec{n}, t + \Delta t) = \frac{1}{N}\sum_{i,j} E_j^+ E_i^- \hat{\pi}_{ij} n_j \Delta t p(\vec{n}, t) + o(\Delta t) \tag{2.32}$$

where the Van Kampen step operators, given any appropriate function f are defined by

$$E_i^{\pm} f(n_1, \cdots, n_{i-1}, n_i, n_{i+1}, \cdots, n_M) = f(n_1, \cdots, n_{i-1}, n_i \pm 1, n_{i+1}, \cdots, n_M).$$

subtracting to both side $p(\vec{n}, t)$ and exploiting the regularization condition (2.29), we re-write equation (2.32)

$$\begin{aligned} p(\vec{n}, t + \Delta t) - p(\vec{n}, t) &= \frac{1}{N}\sum_{i,j} E_j^+ E_i^- \hat{\pi}_{ij} n_j \Delta t p(\vec{n}, t) + \\ &- \frac{1}{N}\sum_{i,j} \hat{\pi}_{ji} n_i \Delta t p(\vec{n}, t) + o(\Delta t)\end{aligned}$$

and finally divide both sides by Δt we take the continuous limit $\Delta t \to 0$ and derive the Master equation

$$\frac{\partial p}{\partial t} = \frac{1}{N}\sum_{i,j} E_j^+ E_i^- \hat{\pi}_{ij} n_j p(\vec{n}, t) - \frac{1}{N}\sum_{i,j} \hat{\pi}_{ji} n_i p(\vec{n}, t) \tag{2.33}$$

At this point we solve the "empty node" issue by extending the probability distribution range to $\mathbb{Z}^M$ and setting the boundary condition

$$p(\vec{n}, t) = 0 \quad \text{if } |\vec{n}| \neq N. \tag{2.34}$$

2.2.2 Exact solution and stationary distribution

The exact solution of equation (2.33) can be computed using the expansion eigenvectors $\vec{v}^\lambda$ of eigenvalue λ of the regularized stochastic matrix $\hat{\pi}$. We recall that, when $\hat{\pi}$ formalizes the transition rates of a connected network, it has a left eigenvector equal to $(1, \cdots, 1)$ of egenvalue 1. The corresponding right eigenvector $\vec{v}$ is the stationary solution of (2.33) with $N = 1$ if properly normalized ($\sum_i v_i = 1$). The remaining eigenvectors (that in principle can have complex entries) have value $|\lambda| < 1$ and satisfy their entries sum up to 0

$$\sum_i v_i^\lambda = 0 \ \ \forall \lambda \neq 1$$

Lemma Given any eigenvalue λ of the transition matrix $\hat{\pi}$, each function of the form

$$f(\vec{n}, t) = e^{-\beta t} \prod_{k=1}^{M} \frac{(v_k^\lambda)^{n_k}}{n_k!}$$

is a solution of Master equation (2.33) if β is defined by

$$\beta = (1 - \lambda) \tag{2.35}$$

Proof

By direct substitution, the left-hand side of (2.33) becomes $-\beta f(\vec{n}, t)$, while the first term on the right-hand side reads

$$\begin{aligned}
&= \frac{1}{N} \sum_{i,j} E_j^+ E_i^- \hat{\pi}_{ij} n_j e^{-\beta t} \prod_{k=1}^{M} \frac{(v_k^\lambda)^{n_k}}{n_k!} \\
&= \frac{1}{N} \sum_{i,j} \hat{\pi}_{ij} (n_j + 1) e^{-\beta t} \frac{(v_i^\lambda)^{n_i - 1}}{(n_i - 1)!} \frac{(v_j^\lambda)^{n_j + 1}}{(n_j + 1)!} \prod_{k \neq i,j} \frac{(v_k^\lambda)^{n_k}}{n_k!} \\
&= \frac{1}{N} \sum_{i,j} \hat{\pi}_{ij} n_i \frac{v_j^\lambda}{v_i^\lambda} f(\vec{n}, t)
\end{aligned}$$

gathering all the terms of the master equation right-hand side we verify

$$\begin{aligned}
&= \frac{1}{N} \sum_{i,j} \hat{\pi}_{ij} n_i \frac{v_j^\lambda}{v_i^\lambda} f(\vec{n}, t) - \frac{1}{N} \sum_{i,j} \hat{\pi}_{ji} n_i f(\vec{n}, t) \\
&= \frac{1}{N} \sum_{i,j} \left[\frac{v_j^\lambda}{v_i^\lambda} \hat{\pi}_{ij} - \hat{\pi}_{ji} \right] n_i f(\vec{n}, t) \\
&= \frac{1}{N} \sum_{i} \frac{1}{v_i^\lambda} \left[\sum_j v_j^\lambda \hat{\pi}_{ij} - v_i^\lambda \hat{\pi}_{ji} \right] n_i f(vecn, t) \\
&= \frac{1}{N} \sum_{i} \frac{1}{v_i^\lambda} \left[\lambda v_i^\lambda - v_i^\lambda \right] n_i f(\vec{n}, t) \\
&= \frac{1}{N} \sum_{i} n_i \left[\lambda - 1 \right] f(\vec{n}, t) \\
&= -\beta f(\vec{n}, t)
\end{aligned}$$

that is the left-hand side of (2.33) after substitution. □.

We can therefore write the general solution of the master equation (2.33)

$$p(\vec{n},t) = N! \sum_{\lambda} e^{-\beta t} c_\lambda \prod_{k=1}^{M} \frac{(v_k^\lambda)^{n_k}}{n_k!} \tag{2.36}$$

with the constant $c_1 = 1$ in order to satisfy the normalizing condition

$$\sum_{|\vec{n}|=N} p(\vec{n},t) = 1 \qquad |\vec{n}| = \sum_i n_i \tag{2.37}$$

the solution (2.35) converges toward the stationary distribution p_s as $t \to \infty$

$$p_s(\vec{n}) = N! \prod_{k=1}^{M} \frac{v_k^{n_k}}{n_k!} \tag{2.38}$$

which is a multinomial distribution with marginal distributions having average and variances respectively

$$\langle n_i \rangle = N v_i, \qquad \sigma^2(n_i) = N(1-v_i)v_i$$

When $v_i \ll 1$ (a condition easy to verify in network of big size) the marginal distribution on the nodes is approximated by a Poisson distribution

$$p_i(k) = \frac{\mu_i^k}{k!} e^{-\mu_i} \quad \text{where} \;\; N v_i = \mu_i \tag{2.39}$$

From this result one, besides the average solution, the value of the variance in each node $(\sigma^2(n_i) = N v_i(1-v_i))$ implies that the expected fluctuations are of order $O(\sqrt{N})$, a typical result of the law of large numbers. We look at the covariance matrix Σ_{ij}

$$\begin{aligned} \Sigma_{ij} &= cov(n_i,n_j) = \sum_{|\vec{n}=N|} n_i n_j \prod_k \frac{v_k^{n_k}}{n_k!} - N^2 v_i v_j \\ &= N(N-1) v_i v_j - N^2 v_i v_j = (\delta_{ij} - v_i) v_i N \end{aligned} \tag{2.40}$$

which implies a negative correlation among the states, as expected.

Remark Restrictions (2.29) on the transition probability together with the constraint on the number of particles are strong assumptions for the random walk dynamics. Indeed the only real issue that could arise would be the "empty node problem"; in order to have a reasonable description of the phenomenon we need to quantify the probability to find an empty node. At stationary state we get

$$p_s(n_i = 0) = N! \sum_{|\vec{n}|=N} \prod_{k \neq i}^{M} \frac{v_k^{n_k}}{n_k!} = (1-v_i)^N = \left(1 - \frac{\bar{n}_i}{N}\right)^N \approx e^{-\bar{n}_i} \tag{2.41}$$

which decreases exponentially as N increases.

2.2.3 The Macroscopic equation

Assuming now N big enough to leave a very small probability for a node to be found empty at any step of the process, and expecting [56], [60] the probability distribution of $\vec{n}$ to have a width of $N^{1/2}$, we operate the classical Linear Noise Approximation (LNA) transformation. In the sequel $\vec{\phi}$ is the deterministic particle concentration function, that can be proved to be the limit as $N \to \infty$ of the stochastic concentration $\vec{n}/N$. In this way we do not need to assume that the function $\vec{\phi}$ satisfies any particular differential equation; if we simply choose it to follow the peak of the distribution as it evolves in time, then the equation it satisfies will emerge. Usually the LNA technique is exploited to study the fluctuations of a stochastic model; however in this special case we use the LNA technique only for the definition of the macroscopic equation since the exact solution of the Master equation (2.33) is already known. In order to perform the LNA of equation (2.33) we rescale time $t = N\tau$ so that it becomes

$$\frac{\partial p}{\partial \tau} = N\left(\sum_{i,j} E_i^- E_j^+ \hat{\pi}_{ij} \frac{n_j}{N} p(\vec{n},\tau) - \sum_{i,j} \hat{\pi}_{ji} \frac{n_i}{N} p(\vec{n},\tau)\right) \tag{2.42}$$

We define the new random variable $\vec{\xi}$ accounting for the fluctuations of particles number on each node so that

$$\vec{n} = N\vec{\phi} + N^{1/2}\vec{\xi} \tag{2.43}$$

with probability distribution Π so that

$$p(\vec{n},\tau) = p(N\vec{\phi} + N^{1/2}\vec{\xi},\tau) = \Pi(\vec{\xi},\tau) \tag{2.44}$$

In order to perform the linear noise approximation we also expand the Van Kampen step operators in equation (2.42) excluding $o(N^{-1})$ terms

$$E_i^- E_j^+ = 1 + \frac{1}{\sqrt{N}}\left(\frac{\partial}{\partial \xi_j} - \frac{\partial}{\partial \xi_i}\right) + \frac{1}{2N}\left(\frac{\partial^2}{\partial \xi_i^2} + \frac{\partial^2}{\partial \xi_j^2} - 2\frac{\partial^2}{\partial \xi_i \partial \xi_j}\right) \tag{2.45}$$

The left-hand side of equation (2.42) becomes:

$$\frac{\partial p}{\partial \tau} = \frac{\partial \Pi}{\partial \tau} + \sum_k \frac{\partial \xi_k}{\partial \tau}\frac{\partial \Pi}{\partial \xi_k} = \frac{\partial \Pi}{\partial \tau} - N^{1/2}\sum_k \frac{\partial \phi_k}{\partial \tau}\frac{\partial \Pi}{\partial \xi} \tag{2.46}$$

where the last term comes form the fact that we assume $\partial n_i/\partial \tau = 0$ when the system is at equilibrium. We now substitute quantites (2.44, 2.45, 2.46) into equation (2.42) and we compare the expressions having the same order in the expansion. For the terms relative to $N^{1/2}$ we obtain:

$$-\sum_k \frac{\partial \phi_k}{\partial \tau}\frac{\partial \Pi}{\partial \xi_k} = \sum_{i,j} \hat{\pi}_{ij}\phi_j\left(\frac{\partial \Pi}{\partial \xi_j} - \frac{\partial \Pi}{\partial \xi_i}\right) \tag{2.47}$$

that holds if $\vec{\phi}$ satisfies the macroscopic equation

$$\dot{\phi}_i = \sum_j \hat{\pi}_{ij}\phi_j - \sum_j \hat{\pi}_{ji}\phi_i \tag{2.48}$$

Proof

we show 2.48 ⇒ 2.47. After a change of index (2.48) reads:

$$\begin{aligned}
\sum_j \frac{\partial\phi_j}{\partial\tau}\frac{\partial\Pi}{\partial\xi_j} + \sum_{i,j}\hat{\pi}_{ij}\phi_j\left(\frac{\partial\Pi}{\partial\xi_j} - \frac{\partial\Pi}{\partial\xi_i}\right) &= 0\\
\sum_j \frac{\partial\phi_j}{\partial\tau}\frac{\partial\Pi}{\partial\xi_j} + \sum_{i,j}\hat{\pi}_{ij}\phi_j\frac{\partial\Pi}{\partial\xi_j} - \sum_{i,j}\hat{\pi}_{ij}\phi_j\frac{\partial\Pi}{\partial\xi_i} &= 0\\
\sum_j \frac{\partial\phi_j}{\partial\tau}\frac{\partial\Pi}{\partial\xi_j} + \sum_{i,j}\hat{\pi}_{ij}\phi_j\frac{\partial\Pi}{\partial\xi_j} - \sum_{i,j}\hat{\pi}_{ji}\phi_i\frac{\partial\Pi}{\partial\xi_j} &= 0\\
\sum_j \frac{\partial\Pi}{\partial\xi_j}\left(\frac{\partial\phi_j}{\partial\tau} + \sum_i\hat{\pi}_{ij}\phi_j - \sum_i\hat{\pi}_{ji}\phi_i\right) &= 0
\end{aligned} \tag{2.49}$$

where the terms in the round brackets are all null if the macroscopic equation (2.48) holds. We can therefore say that if equation (2.48) holds then equation (2.47) is verified. □

Equation (2.48) written in matrix notation becomes

$$\frac{d\vec{\phi}}{dt} + L\vec{\phi} = 0 \tag{2.50}$$

where we recall the Laplacian matrix of the system

$$L_{ij} = d_i\delta_{ij} - \hat{\pi}_{ij} \qquad d_i = \sum_j \hat{\pi}_{ji} \tag{2.51}$$

The stationary solution $\vec{\phi}_s$ corresponds to the unique eigenvector of null eigenvalue of matrix L.

Remark The macroscopic equation (2.48) obtained through the LNA is formally equivalent to the microscopic mean-field equation (2.30). However the two equations have a different physical meaning and the connection between them is not straightforward. Mirzaev and Gunawardeena ([51]) in their exaustive review about Laplacian dynamics notice that the diffusion of a substance on a graph can be seen as a Master equation when the substance represents an average probability to find a particle a in node i at time t.

For sake of completeness, we conclude the LNA approximation: equalizing the N^0 terms, neglecting higher orders and after some algebra, we obtain a linear Fokker-Plank equation

$$\begin{aligned}
\frac{\partial\Pi}{\partial\tau} &= -\vec{\nabla}_\xi^T\cdot(-L)\cdot\vec{\xi}\Pi + \frac{1}{2}\vec{\nabla}_\xi^T\cdot\left[LD_{\phi_s} + (LD_{\phi_s})^T\right]\cdot\vec{\nabla}_\xi\Pi\\
\frac{\partial\Pi}{\partial\tau} &= -\vec{\nabla}_\xi^T\cdot A\cdot\vec{\xi}\Pi + \frac{1}{2}\vec{\nabla}_\xi^T\cdot B\cdot\vec{\nabla}_\xi\Pi
\end{aligned} \tag{2.52}$$

where we highlight the drift matrix $A = -L$ and the diffusion matrix $B = LD_{\phi_s} + (LD_{\phi_s})^T$; here $\vec{\nabla}_\xi = (\partial/\partial\xi_1, \cdots, \partial/\partial\xi_M)^T$ and $D_{\phi_s} = \mathrm{diag}(\vec{\phi_s})$. Equation (2.52) has a Gaussian stationary distribution with average given by the macroscopic solution $\vec{\phi}_s$ and covariance matrix Σ being the solution of the Lyapunov equation:

$$A\Sigma + \Sigma A^T + B = 0. \tag{2.53}$$

Solution of equation (2.53) is normally non-trivial and in most cases numerical approaches are needed [60]. As previously mentioned the linear noise approximation approach can be exploited to solve with a good approximation the Master equation fluctuations and in this context would be redundant since the exact solution can be obtained from direct calculation. However in more general cases, for example if one models the transitions among states as chemical reactions an approach as LNA is fundamental.

2.2.4 Maximal entropy principle

We now show that in the network-based random walk the stationary distribution is a maximal entropy solution. We underline that in equation (2.33) the nodes are not independent since the number of particles is limited to N and of course the average value of particles in each node is finite

$$\bar{n}_i = \sum_{|\vec{n}|=N} n_i p(\vec{n}).$$

we use the last M equations as a constraint to the Gibbs Entropy

$$S = -\sum_{|\vec{n}|=N} p(\vec{n}) \ln(\omega(\vec{n})p(\vec{n})) - \sum_i \mu_i \sum_{|\vec{n}|=N} n_i p(\vec{n}) \tag{2.54}$$

where μ_i are the Lagrangian multipliers and $\omega(\vec{n})$ is a coefficient describing the statistical weight of the network state $\vec{n}$ depending on the microscopical dynamics

$$\omega(\vec{n}) = \prod_{i=1}^{M} n_i! \qquad |\vec{n}| = N \tag{2.55}$$

Namely $\omega(\vec{n})$ associates a weight to the state $\vec{n}$ on the basis of how many ways the random walkers can reach such a state. Once we perturb the probability distribution we obtain

$$\delta S = -\sum_{|\vec{n}|=N} \ln(\omega(\vec{n})p(\vec{n}))\delta p(\vec{n}) - \sum_i \mu_i \sum_{|\vec{n}|=N} n_i \delta p(\vec{n}) = 0 \tag{2.56}$$

that leads to

$$\ln(\omega(\vec{n})p(\vec{n})) = \sum_i \mu_i n_i \Rightarrow p(\vec{n}) \propto \frac{1}{\omega(\vec{n})} e^{-\sum_i \mu_i n_i}$$

and we identify $\mu_i = -\ln(\bar{n}_i/N)$. We showed that the stationary distribution is a Maximal Entropy solution.

2.2.5 Detailed balance

In this paragraph we derive the detailed balance condition assuming independency among the nodes of the network. The probability distribution dynamics for a single node marginal distribution $p_i(k,t)$ (the probability to find k particles in node i at time t), with this assumption becomes

$$\frac{\partial p_i(k,t)}{\partial t} = \sum_k \sum_j \hat{\pi}_{ij} \frac{k}{N} p_j(k,t) - \sum_k \sum_j \hat{\pi}_{ji} \frac{k}{N} p_i(k,t) \tag{2.57}$$

This equation is not capable of explaining the transient states of the process; however it converges for any node to its marginal stationary poisson distribution $p_i(k) = \frac{(Nv_i)^k}{k!} e^{-Nv_i}$, where $\vec{v} = \vec{p}_s$ is the eigenvector of eigenvalue 1 of the stochastic matrix π. Plugging it in in (2.57) we get

$$\begin{aligned} 0 &= \sum_k \sum_j \hat{\pi}_{ij} \frac{k}{N} \frac{(Nv_j)^k}{k!} e^{-Nv_j} - \sum_k \sum_j \hat{\pi}_{ji} \frac{k}{N} \frac{(Nv_i)^k}{k!} e^{-Nv_i} \\ &= \sum_j \hat{\pi}_{ij} v_j e^{-Nv_j} \sum_k \frac{(Nv_j)^{k-1}}{k-1!} - \sum_j \hat{\pi}_{ji} v_i e^{-Nv_i} \sum_k \frac{(Nv_i)^{k-1}}{k-1!} \\ &= \sum_j \hat{\pi}_{ij} v_j - \sum_j \hat{\pi}_{ji} v_i \\ &= \sum_j \hat{\pi}_{ij} v_j - v_i \end{aligned}$$

where the last quantity is null by definition of $\vec{v}$ and $\hat{\pi}$. Equation (2.14) can also be written in a more compact form

$$\frac{\partial p_i(k,t)}{\partial t} = \sum_j J_{ij}$$

where the currents (or fluxes) J_{ij} in the link $j \to i$ are defined as

$$J_{ij} = \sum_k \hat{\pi}_{ij} \frac{k}{N} p_j(k,t) - \sum_k \hat{\pi}_{ji} \frac{k}{N} p_i(k,t)$$

In this perspective is easy to see that detailed balance implies that the probability currents $J_{ij} = 0$ for for each couple of connected nodes of the network in the stationary state. This condition implies that the stationary state satisfies the condition

$$\frac{\hat{\pi}_{ij}}{\hat{\pi}_{ji}} v_j = v_i \tag{2.58}$$

The detailed balance condition derives from the node-independency assumption. In this context all internal forces acting in the system are conservative so that one can define the potential $V_i = -\ln v_i$; such quantity is positive by definition. The detailed balance condition now reads

$$V_j - V_i = \ln \frac{\hat{\pi}_{ij}}{\hat{\pi}_{ji}} \qquad i \neq j$$

so that the left-hand side can be interpreted as potential energy, while the right-hand side is the work to move one particle through the link $j \to i$. Under the detailed balance condition (2.58) the internal forces are therefore conservative: the work to move a particle from any node to another one doesn't depend on the path.

2.2.6 Stochastic matrices and detailed balance

Now we focus on the properties of stochastic matrices in relation to the detailed balance condition. We consider a stochastic matrix π with correspondent stationary state e^{-V}; here we omit the hat over the entries of the matrix π. Given any couple of vectors $\vec{u}, \vec{w} \in \mathbb{R}^M$ we define a scalar product as

$$\vec{w} \cdot \vec{u} = \sum_i w_i e^{V_i} u_i \tag{2.59}$$

that is evidently defined on the basis of the stationary distribution e^{-V_i}. We define the adjoint matrix π^* of the stochastic matrix π with respect to the scalar product $\cdot$ as

$$\pi^*_{ij} = e^{-V_i} \pi_{ji} e^{V_j} \qquad \forall i, j \tag{2.60}$$

We state the following

Lemma Given a stochastic matrix π

i. The adjoint matrix π^* is still stochastic with the same stationary state e^{-V}.

ii. Given any couple of vectors $\vec{u}, \vec{w} \in \mathbb{R}^M$ the adjoint matrix π^* satisfies the equation

$$\vec{w} \cdot \pi \vec{u} = \pi^* \vec{w} \cdot \vec{u}$$

where the scalar product is defined as in (2.59).

Proof

i. We consider the adjoint matrix $\pi^*_{ij} := e^{-V_i} \pi_{ji} e^{V_j}$; we verify the stocasticity of such matrix:

$$\sum_j \pi^*_{ij} = \sum_j e^{-V_i} \pi_{ji} e^{V_j} = 1$$

We verify by direct substitution that e^{-V} is the stationary solution

$$\pi^* e^{-V} = e^{-V} \pi^T = \sum_j e^{-V_j} \pi_{ji} = e^{-V} \qquad \square$$

ii.

$$\begin{aligned} \vec{w} \cdot \pi \vec{u} \;\; &= \;\; \sum_k w_k e^{V_k} (\pi \vec{u})_k = \sum_k w_k e^{V_k} \sum_j \pi_{kj} u_j \\ & \;\; \sum_{k,j} e^{-V_j} \pi_{kj} e^{V_k} w_k e^{V_j} u_j = \pi^* \vec{w} \cdot \vec{u}. \qquad \square \end{aligned}$$

Thanks to this lemma we see that the detailed balance condition (2.58) is equivalent to the self-adjoint condition with respect to the scalar product (2.59):

$$\pi_{ij} = e^{-V_i} \pi_{ji} e^{V_j} \qquad \forall i, j \tag{2.61}$$

We also remark that if we use the change of base $\vec{u}' = e^{V/2}\vec{u}$, the scalar product (2.60) gains the standard form

$$\vec{w} \cdot \vec{u} = \sum_k e^{-V_k/2} w'_k e^{V_k} e^{-V_k/2} u'_k = \sum_k w'_k u'_k \tag{2.62}$$

and the matrix π is normalized as

$$\pi' = e^{V/2} \pi e^{-V/2} \tag{2.63}$$

It is straightforward to notice that if π is self adjoint then π' is symmetric:

$$\pi'_{ij} = e^{V_i/2} \pi_{ij} e^{-V_j/2} = e^{V_i/2} \left(e^{-V_i} \pi_{ji} e^{V_j} \right) e^{-V_j/2} = e^{V_j/2} \pi_{ji} e^{-V_i/2} = \pi'_{ji}$$

In addition the state $e^{-V/2}$ is the stationary distribution of the normalized matrix (2.63). We use these results for identifying stochastic matrices that fit Master equations satisfying the detailed balance condition. We basically prove that the contrary of the last remark holds as well:

Lemma Given any stochastic matrix S with positive entries and an eigenvector $e^{-V/2}$ with eigenvalue 1, then the matrix found with the transformation:

$$\pi = e^{-V/2} S e^{V/2} \tag{2.64}$$

is stochastic with stationary state e^{-V}. In addition all the eigenvalues are real numbers smaller than 1 and satisfies the detailed balance condition (2.58).

Proof we prove the stochasticity of the matrix π by direct calculation:

$$\sum_i \pi_{ij} = \sum_i e^{-V_i/2} S_{ij} e^{V_j/2} = e^{-V_i/2} \sum_i S_{ji} e^{V_j/2} = e^{-V_i/2} e^{V_i/2} = 1$$

where in the second passage we used that S is symmetric. The stationary state of π is e^{-V}:

$$\sum_j \pi_{ij} e^{-V_j} = \sum_j e^{-V_i/2} S_{ij} e^{V_j/2} e^{-V_j} = \sum_j e^{-V_i/2} S_{ij} e^{-V_j/2} = e^{-V_i}$$

the detailed balance condition is satisfied by definition

$$\frac{\pi_{ij}}{\pi_{ji}} = \frac{e^{-V_i/2} S_{ij} e^{V_j/2}}{e^{-V_j/2} S_{ij} e^{V_i/2}} = e^{-(V_i - V_j)} \tag{2.65}$$

Therefore the matrix π_{ij} has all real eigenvalues □.
We now generalize the previous lemma, characterizing all the stochastic matrices π, satisfying detailed balance (2.58):

Theorem Consider a stochastic matrix π that is symmetric with respect to a real positive defined quadratic form

$$(\vec{u}, \vec{w}) = \sum_{i,j} u_i h_{ij} w_j$$

then π satisfies detailed balance (2.58).

Proof We start by the symmetric condition

$$\pi H = H \pi^T \tag{2.66}$$

where H is a symmetric positive defined matrix. Then H is a distributed stationary state:

$$\sum_{i,j} \pi_{ij} h_{jk} = \sum_{i,j} h_{ij} \pi_{kj} = \sum_{j} \pi_{kj} \left(\sum_{i} h_{ij} \right)$$

where

$$e^{-V_i} := \sum_j h_{ji}$$

and detailed balance reads formally as (2.65) □.

When we say that H ia a distributed stationary state, we mean that its actual stationary state e^{-V} has components equal to the sums of the columns of H. This implies that the term $\pi_{ij} g_{jk}$ can be seen as the flow on the link $j \rightarrow i$ thanks to the k-th component of the distributed stationary state H and summing up all such contributions

$$\sum_j \hat{\pi}_{ij} h_{jk}$$

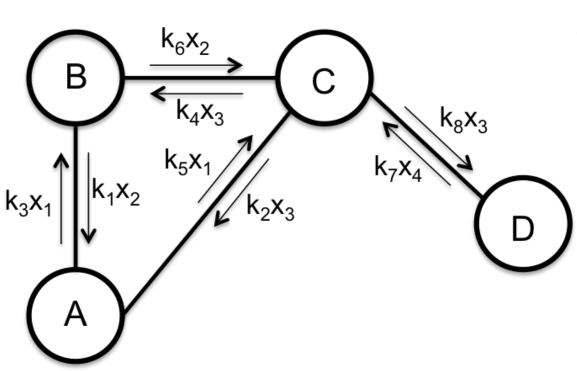
one gets the incoming flow on the node i associated to the k-th component of the stationary state. In this context one can interpret condition (2.66) as another formulation of detailed balance.

2.2.7 Biochemical networks

So far we developed the random walk on a network, and under choosen assumptions we discussed the phenomenon accurately. Now, we want to propose the discussed random walk model as a particular case of Biochemical network framework that is discussed exensively in the work by Elf and Ehremberg [60]. The idea is that, given a biological network, the transitions between states are modelled by linear or non-linear functions of the states. The reason why this generalization could be important relies on the fact that intra or inter omics exchanges could be poorly described by a random walk. Of course the random walk model is interesting for many reasons - not last its applicability - but it remains fundamentally at the base of many computational tools that can be exploited on the interactome to perform several tasks as we will show in the sequel.

In the perspective that a more accurate mathematical description of the intra-cellular biological process should lead to better performing statistical tools, we can think of the nodes of the network as intracellular species interacting with each other according to dependent or independent exchanges of individuals. For example one could allow the birth and the death of individual particles leading to a model that would be the generalization of a One-step process in M variables, where M is the size of the network. Another choice could be to model the exchanges between states as chemical reactions with a given transition probability and stochiometry.

We now consider the array of random varibles $\vec{n}$ with a constant volume of particles $N = \sum_i n_i(t), \forall t$ with each state having discrete range. The most general form of



$$S = \begin{array}{c|cccccccc} & k_1 & k_2 & k_3 & k_4 & k_5 & k_6 & k_7 & k_8 \\ \hline [A] & 1 & 1 & -1 & 0 & -1 & 0 & 0 & 0 \\ [B] & -1 & 0 & 1 & 1 & 0 & -1 & 0 & 0 \\ [C] & 0 & -1 & 0 & -1 & 1 & 1 & 1 & -1 \\ [D] & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{array}$$

$$L = \begin{array}{c|cccc} & [A] & [B] & [C] & [D] \\ \hline [A] & 1 & -1/2 & -1/3 & 0 \\ [B] & -1/2 & 1 & -1/3 & 0 \\ [C] & -1/2 & -1/2 & 1 & -1 \\ [D] & 0 & 0 & -1/3 & 1 \end{array}$$

$$f(x) = [k_1x_2, k_2x_3, k_3x_1, k_4x_3, k_5x_1, k_6x_2, k_7x_4, k_8x_3]^T$$

Figure 2.3: **Stochiometrics and Laplacian** We explicitely give the toy network of example 1 the numerical quantites treated. L is the Laplacian matrix of the closed system; S and $\mathbf{f}$ are respectively the stochiomeric matrix and the macroscopic transition rates for the closed system

Master equation on a network written in compact form:

$$\frac{\partial P(\vec{n},t)}{\partial t} = N \sum_{j=1}^{R} \left[\prod_{i=1}^{M} E^{-S_{ij}} - 1 \right] f_j(\vec{n}/N) P(\vec{n},t) \tag{2.67}$$

where we are considering R possible exchanges with transition rates $f_1, \cdots, f_R$, S is a M × R matrix describing in each column a different reaction f_j; such matrix is called S since in the context of biochemical networks it would be the Stoichiometric matrix defining the macroscopic exchanges between states. In equation (2.13) we introduce the Van Kampen step operator $E^{-S_{ij}}$ that is defined by its action on any function g of the state $\vec{n}$

$$E^{-S_{ij}} g(\vec{n}) = g(n_1, \cdots, n_i - S_{ij}, \cdots n_M)$$

We remark that this is the general case meaning that form (2.67) includes also the case in which exchanges of particles or reactions are dependent from one another so that the products on the right hand side may not commute making solution of (2.13) very difficult. We also underline that the transitions between states can in principle be linear or non-linear functions of the states $\vec{n}$. Of course non-linearity of such transitions increases as well the complexity of the problem.

In the undirected toy network shown in Fig. 2.3 we write down the stochiometric matrix S and the macroscopic transition rates f_j as in (2.3) and we observe that if we choose the k_j to be constant and equal to the reciprocal of the degree from which the reaction starts, the macroscopic equation becomes equivalent to a random walk. More precisely, if we take $k_1 = \frac{1}{2}, k_2 = \frac{1}{3}, k_3 = \frac{1}{2}, k_4 = \frac{1}{3}, k_5 = \frac{1}{2}, k_6 = \frac{1}{2}, k_7 = 1, k_3 = \frac{1}{3}$, at stationary state we get:

$$Sf(\vec{\phi}_s) = L_g G^{-1} \vec{\phi}_s = L\vec{\phi}_s = 0 \Leftrightarrow \vec{\phi}_s = \frac{N}{2a}(g_1, \cdots, g_M)^T$$

where a is the total number of connections and g_i is the degree of node i in the network. Such solution is reached at infinite time starting from any initial distribution of the substance on the network.

With this simple observation (Fig. 2.3) we suggest that the random walk on the network can be seen as a special case of reaction network. The general model (2.67) could be useful for more accurate modelling of exchanges between molecular species in the omics data context. However the following questions remain undiscussed in this work: when and how is it possible to shift from a stochiometric matrix S and associated transition rates describing the exchanges among states to a Laplacian formulation? Or at least when and to which grade of error is it possible to approximate a complex reaction network with a Laplacian dynamics?

2.3 The macroscopic equation as a hydrodynamic model

In the previous sections we defined and studied some stochastic models with the purpose of introducing a theoretical framework to be applied to omic data integration. The underlying assumption is that relations between molecular entities can be modeled through (direct or undirect) networks.

In principle each omic layer has its own way to be modeled and at the state of the art the complexity of the cell is far away to be described with a good approximation with a unique mathematical framework. In this perspective the formulation of a general model like (2.67) could lead to a better description of omic-specific or inter-omic intracellular dynamics. However the formulation of such models is often still far away from the methodology and tools used by applied scientists; the statistical challenges arising in the analysis of an omic (or multi-omic) dataset usually needs a more practical "down to earth" approach since sophisticated mathematical approaches on one hand would sometimes have to rely on a weak or incomplete *a priori* knowledge and on the other hand the concrete goals of biological and medical scientists need fast and simple solutions.

As described in chapter 1 network-diffusion based methods are an example of how more sophisticated mathematical modeling of the data can lead to better results. When we speak about network-diffusion based methods we mean data manipulation methods that diffuse the initial information about molecular entities on a given network that - to a certain degree of accuracy - quantifies the relationships between such molecular entities. Network diffusion methods consists of ad hoc algorithms with parameters that can be tuned depending on the scientists' goal. In chapter 1 we mentioned diffusion kernels [36, 45] and the propagation algorithms [46, 47, 48, 57] as two possible approaches to perform such network diffusion. Now we focus on the second one and we show that the network propagation algorithm can be derived by the master equation model described in the previous sections.

2.3.1 An open source/sink model

The linear Laplacian dynamics is well described and exaustively treated by Mirzaev and Gunawardeena [51]. With Laplacian dynamics we mean the diffusion of a substance on a network having Laplacian matrix L. Such equation - as they suggest - is formally equivalent to the macroscopic equation arising from the master equation described in the previus section (2.50) that we write in compact form:

$$\frac{d\vec{\phi}}{dt} + L\vec{\phi} = 0 \tag{2.68}$$

where we recall the definition the Laplacian matrix of the system

$$L_{ij} = d_i\delta_{ij} - \hat{\pi}_{ij} \qquad d_i = \sum_j \hat{\pi}_{ji}$$

Such equation can be interpreted as a master equation itself: the substance diffusing on the network $\vec{\phi}$ can be seen as the average probability for a particle to be found in a given node at a given time if we introduce the constraint

$$\sum_k \phi_k(t) = 1.$$

We demonstrated that equation (2.68) can be derived as the macroscopic equation (2.48) associated to the general master for a random walk of particles on the network (2.33). If $d_k = 1$ the matrix $\hat{\Pi}$ becomes a stochastic matrix with the stationary solution $\vec{\phi}_s$ being the right eigenvector corresponding to the null eigenvalue of the matrix L. Equation (2.68) describes a closed isolated system in which starting from any initial probability distribution $\vec{\phi}_0 = \vec{\phi}(0)$, the dynamics relaxes to the same stationary solution. In fact since the eigenvalues of L are all positive except the null one, $\vec{\phi}_s$ is unique and attractive.
Introducing a source-sink perturbation to this isolated system we get an open system:

$$\begin{aligned}\frac{d\vec{\phi}}{dt} + L\vec{\phi} - s_0\vec{\pi}_{in} + I\cdot\vec{\pi}_{out}\cdot\vec{\phi} &= 0\\ \frac{d\vec{\phi}}{dt} + (L + I\cdot\vec{\pi}_{out})\cdot\vec{\phi} - s_0\vec{\pi}_{in} &= 0 \qquad (2.69)\end{aligned}$$

where $\vec{\pi}_{in} = (\pi_{10}, \cdots, \pi_{M0})^T$ are the input connections and $\vec{\pi}_{out} = (\pi_{01}, \cdots, \pi_{0M})^T$ are the sink rates from each node. Equation (2.69) can be solved since the correspondent homogeneous system admits a unique solution ($L + I\vec{\pi}_{out}$ is diagonally dominant). The stationary state is given by

$$\vec{\phi}_s = (L + I\cdot\vec{\pi}_{out})^{-1}s_0\vec{\pi}_{in} \tag{2.70}$$

2.3.2 The hydrodynamic interpretation

Starting from equation (2.69) and assuming the sink vector $\vec{\pi}_{out}$ to be a constant vector with entries equal to a first order sink rate γ, we can write in compact form

$$\frac{d\vec{\phi}}{dt} = -L_\gamma\vec{\phi} + s_0\vec{\phi}_0 = 0 \tag{2.71}$$

where $\vec{\phi}_0 := \vec{\pi}_{in}$ is the input array whose i-th entry is 1 if the node i is a source node and 0 otherwise and $L_\gamma := L + I\gamma$. In this case L is the symmetric version of the Laplacian after change of base (2.62). Some interesting biological applications(ref)

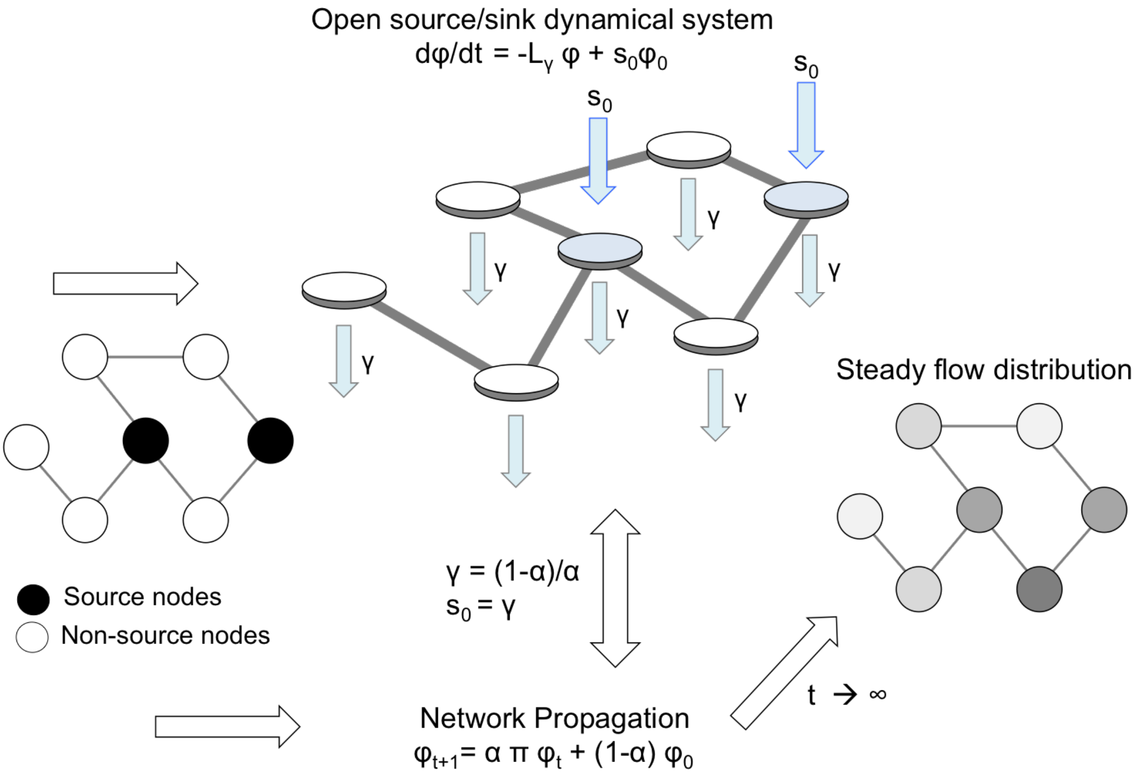


Figure 2.4: **Network Propagation and source/sink model** The Network Propagation algorithm can be used to recover the stationary distribution of the perturbed hydrodynamical system (2.69). Here we use the restrictions on the sources and sinks that we found in literature.

of equation (2.71) are based on its hydrodynamic interpretation. The source nodes are decribed by the M-components vector $\vec{\phi}_0$. Now we assume that the source nodes on the network are thought as valves that open at time $t = 0$ with a constant incoming flow rate s_0. So starting from $t = 0$ the fluid coming from the opened valves spreads throughout the network according to the existing connections among the nodes and exits from the nodes according to the constant sink rate γ. The effect of such a set up leads to a steady flow state where at infinite time the amount of fluid remaining in each node is constant even if it continues to enter the network through the valves and exit from the sinks. In Fig. 2.4 we show a simple visualization of the model.
The hydrodynamic system (2.71) and its generalization (2.69) can be used on a biological network for example for gene prioritization. Another application is the "hot" subnetworks extraction. With the word "hot" we mean those areas of the network that are enriched in sources that usually represent abnormal data information. In this work we deduce the dynamical system (2.71) from the Master equation adapted on the network getting to a general model. In fact equation (2.69) can in principle differenciate the incoming flow rate s_0 from node to node using real inputs $\vec{\pi}_{in}$ as well as the sinks $\vec{\pi}_{out}$.

2.3.3 Connection to the network propagation algorithm

Since for big networks inverting the matrix $L + I \cdot \vec{\pi}_{out}$ is computationally out of reach one can use a numerical approach as shown in figure (2.4); in order to write a forward Euler version of (2.69) we first assume that the sinks $\vec{\pi}_{out}$ are all equal to a positive constant $\pi_{k0} = \gamma$ as well as the sources $s_0 = \gamma$; substituting into (2.69) we get

$$\begin{aligned} \frac{d\vec{\phi}}{dt} &= -(L + I\gamma) \cdot \vec{\phi} + \gamma\vec{\phi}_0 \\ \frac{d\vec{\phi}}{dt} &= -(I - \hat{\pi} + I\gamma) \cdot \vec{\phi} + \gamma\vec{\phi}_0 \\ \frac{d\vec{\phi}}{dt} &= -(I(1+\gamma) - \hat{\pi}) \cdot \vec{\phi} + \gamma\vec{\phi}_0 \qquad (2.72) \end{aligned}$$

where $\vec{\phi}_0 := \vec{\pi}_{in}$ is thought as the input and $\hat{\pi} = I - L$ is the transition matrix; re-parametrizing $\gamma = (1-\alpha)/\alpha$, with $0 < \alpha < 1$,

$$\frac{d\vec{\phi}}{dt} = -\left(I\frac{1}{\alpha} - \hat{\pi}\right) \cdot \vec{\phi} + \frac{1-\alpha}{\alpha}\vec{\phi}_0 \qquad (2.73)$$

so that after multiplying by the parameter α and rescaling time equation (2.73) becomes

$$\frac{d\vec{\phi}}{d\tau} = -\left(I - \alpha\hat{\pi}\right) \cdot \vec{\phi} + (1-\alpha)\vec{\phi}_0 \qquad (2.74)$$

At this point we discretize time using a time step $\Delta\tau$ and re-adapt equation (2.73)

$$\begin{aligned} \frac{\vec{\phi}(\tau + \Delta\tau) - \vec{\phi}(\tau)}{\Delta\tau} &= -\left(I - \alpha\hat{\pi}\right) \cdot \vec{\phi}(\tau) + (1-\alpha)\vec{\phi}_0 \\ \vec{\phi}(\tau + \Delta\tau) - \vec{\phi}(\tau) &= -\left(I - \alpha\hat{\pi}\right) \cdot \vec{\phi}(\tau)\Delta\tau + (1-\alpha)\vec{\phi}_0\Delta\tau \qquad (2.75) \end{aligned}$$

assuming now a unitary time step $\Delta\tau = 1$ so that $\vec{\phi}(\tau + \Delta\tau) = \vec{\phi}_{\tau+1}$ (2.75) and we can re-write equation as

$$\vec{\phi}_{\tau+1} = \alpha\hat{\pi} \cdot \vec{\phi}_\tau + (1-\alpha)\vec{\phi}_0 \qquad (2.76)$$

In equation (2.76) during each iteration each node receives the information from its neighbors, and also retains its initial information and self-reinforcement is avoided. The information is spread according to the transition matrix $\hat{\pi}$. We now demonstrate that algorithm (2.76) converges to the stationary distribution (2.70) that written substituting the parameter α reads

$$\vec{\phi}_s = (1-\alpha)(I - \alpha\hat{\pi})^{-1}\vec{\phi}_0 \qquad (2.77)$$

Proof we demonstrate that (2.76) converges to (2.77) through the power expansion method; starting from ϕ_0 (omitting the vector subscript) and we apply some iterations:

$$\begin{aligned} \phi_1 &= \alpha\hat{\pi}\phi_0 + (1-\alpha)\phi_0, \\ \phi_2 &= \alpha\hat{\pi}\phi_1 + (1-\alpha)\phi_0 \\ &= \alpha\hat{\pi}(\alpha\hat{\pi}\phi_0 + (1-\alpha)\phi_0) + (1-\alpha)\phi_0 \\ &= (\alpha\hat{\pi})^2\phi_0 + (1-\alpha)(\alpha\hat{\pi} + I)\phi_0 \\ &= (\alpha\hat{\pi})^2\phi_0 + (1-\alpha)((\alpha\hat{\pi}) + (\alpha\hat{\pi})^0)\phi_0 \end{aligned}$$

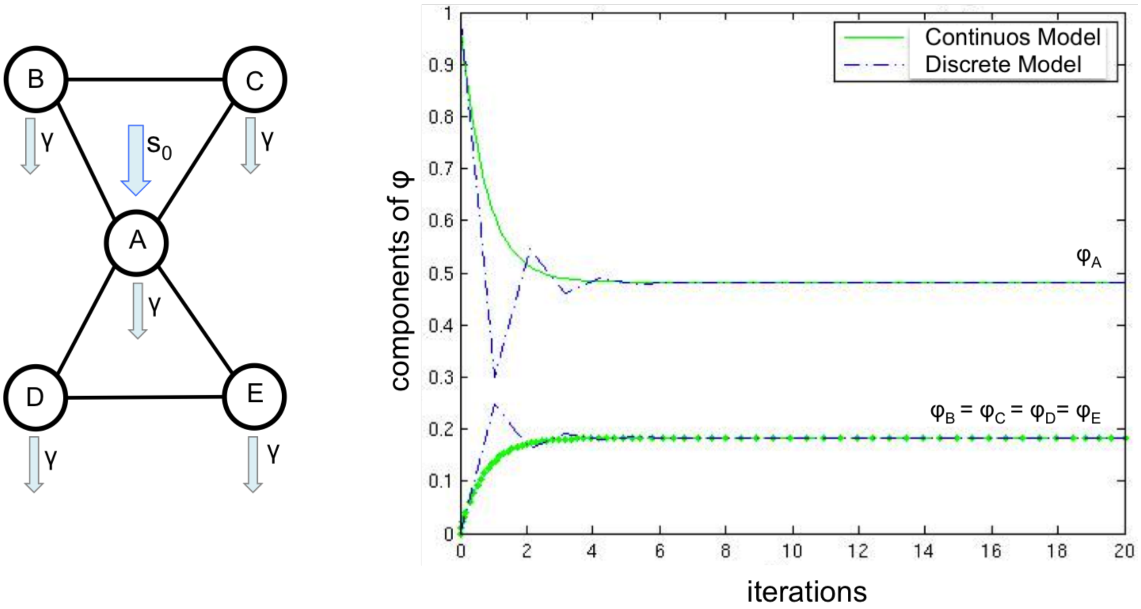


Figure 2.5: **Comparison between Network Propagation and source/sink model** Numerical comparison on a simple undirected toy network with initial condition $\vec{\phi}_0 = (1,0,0,0,0)^T$ where we put the source in the node A. In green we perform the numerical solution of the dynamical system (2.71) seen as distinct components and we compare it with the network propagation algorithm (2.76) in blue.

Iterating this procedure at step t we get:

$$\phi_t = (\alpha\hat{\pi})^t\phi_0 + (1-\alpha)\sum_{i=0}^{t-1}(\alpha\hat{\pi})^i\phi_0,$$

Since $0 < \alpha < 1$ and the eigenvalues of $\hat{\pi}$ are in $[-1;1]$, if the transition matrix is properly taken (*). when we take the limit for $t \to \infty$ we get:

$$\phi_s = (1-\alpha)(I-\alpha\hat{\pi})^{-1}\phi_0$$

that is (2.77). □
(*) The previous demonstration is valid for all transition matrices $\hat{\pi}$ whose eigenvalues have module smaller than 1. We remark that several choices of transition matrices are acceptable. Starting from an undirected graph with adjacency matrix A and diagonal degree matrix G, the most interesting choices are the stochastic matrix $\hat{\pi}' = AG^{-1}$ or the symmetric matrix $G^{-1/2}AG^{-1/2}$. The last choice comes from the change of base (2.62) from the stochastic matrix $\hat{\pi}$ to the symmetric matrix $\hat{\pi}' = G^{-1/2}\hat{\pi}G^{1/2}$. For clarification about matrix notation see Fig. 2.2.
Choosing $\hat{\pi}'$, equation (2.76) is a known network diffusion algorithm known as "network propagation" [57]. This method is successfully exploited by Hofree et al. [46] in their network based approach applied to somatic mutation profiles. The physical interpretation of such a diffusive algorithm inherits some helpful concepts from the associated physical model. The parameter α describes how much the fluid is free to propagate in the network versus how much it tends to remain in the source nodes. We demonstrated, under simplified assumptions, that the network propagation algorithm is an Euler forward discretization of the macroscopic equation derived from the random walk on the network. In Fig. 2.5 we see how the network propagation

algorithm helps find the stationary solution as it rapidly converges to the stationary distribution of the source/sink model.

We remark that introducing the sources and the sinks gives rise to an open system, where the amount of fluid in the network is not conserved. In fact the stationary solution (2.70, 2.77) is a constant-flow solution where the incoming flow stabilizes at infinite time with the outgoing one. The only possibility to allow the fluid conservation is the introduction of a source/sink node that extends the network so that $\vec{\phi}$ can be still interpreted as a probability distribution with the constraint

$$\sum_k \phi_k(t) = 1.$$

and equation (2.68) can be thought as an actual Master equation. The source/sink node extension and its implications are developed in chapter 4.

The network propagation algorithm is exploited mainly in the next chapter where we describe an application to differentially active module discovery. However the general master equation model (2.68, 2.69) from which it is derived will be exploited in chapter 4 for perturbation analysis and control theory.

Part III
Applications

Chapter 3

Network-diffusion based analysis of high-throughput data

This chapter describes the application of the hydrodynamic model described in the previous chapter to the analysis of omics data. In particular we exploit the connection between the hydrodynamic model and the network propagation algorithm to define novel omics measures based on the network diffusion of the omic information on the network. We apply such techniques to the discovery of differentially enriched modules that consist of network regions enriched in statistics derived from different types of omics datasets. Supplementary information and further insights are available on the published paper *"Network diffusion-based analysis of high-throughput data for the detection of differentially enriched modules"* [61].

3.1 Biological networks and diffusion techniques

Cellular functions are carried out by modules of interacting molecular entities [62]. Complex intracellular circuits can be modeled as networks in which vertices are molecular entities and links are (direct and indirect) interactions among entities. According to the so-called local hypothesis, functional similarity is related to network proximity and, in line with it, the molecular entities involved in the same disease have an increased tendency to interact with each other [6]. This knowledge, in combination with the growing availability of molecular interactions data, offers the opportunity to develop computational approaches that use network proximity as a tool to predict molecular species function and disease association [44, 63].

More generally, the definition of the network regions associated to biological functions and diseases is a major goal in systems biology[6, 64]. Several integrative approaches, which jointly analyse interactions and molecular profiles, have been proposed [46, 47, 65, 66, 67, 68, 69] and were recently classified into four broad categories: identification of active modules, identification of conserved modules across species, identification of differential modules and identification of composite modules [70]. However, this task is still an open challenge in bioinformatics research. First of all, the size of biological networks makes the search for subnetworks time-consuming. Secondly, technological biases in high-throughput approaches for interaction detection and molecular profiling can compromise analyses accuracy. Thirdly, our biological knowledge is still limited: just to mention two relevant examples, according to recent estimates, only the 10% of protein-protein interactions (PPIs) may be known [71] and while more than half of all proteins are glycosylated, knowledge about the glycosylation process is still limited [72]. Another challenging aspect is that while topological communities often represent functional modules, they do not overlap with disease modules: therefore, the search for disease subnetworks can not be faced using only community detection methods [6, 65].

Recently, diffusion-based approaches, which simulate the diffusion of a quantity throughout a network in order to calculate a global measure of network proximity, have been successfully proposed in several applications, taking advantage of the local hypothesis. A few examples are the association of genes and protein complexes with diseases [57], the stratification of tumour mutations [46], the identification of biomarkers in genome-wide studies [73, 74] and the study of virus-host molecular interactions [75, 76].

Examples of diffusion-based bioinformatics tools include NBS [46], HotNet [47], TieDie [67], ResponseNet [68], RegMod [69] and stSVM [77]. NBS [46] smooths somatic mutations profiles and than uses neetwork-based non-negative matrix factorisation to stratify subjects. Hotnet [47] uses summary statistics derived from somatic mutations as input for a diffusion process in order to identify active network regions. Hotnet2 [66] uses an insulated heat diffusion that, roughly speaking, comes from a non-symmetrical normalization of the network's adjacency matrix which correct for vertex degree, thus intrinsically reducing the weight of hubs. In fact, the output of several diffusion-based methods shows a dependency on vertex degree [65, 74, 76]. Hotnet2 integrates the diffusion matrix, which contains topological information, with somatic mutations and, then, identifies hot subnetworks selecting high scoring links. The significance of the number and size of the subnetworks is calculated using a two-stage statistical test. TieDie (Tied Diffusion Through Interacting Events) [67] and ResponseNet [68] use two different approaches to find the subnetwork that connects two sets (sources and targets) of network vertices, which can represent genomic perturbations and gene expression variations. TieDie [67] uses a diffusion approach to find a subnetwork of sources, targets and (predicted) linkers that are "logically consistent" in relation to their molecular profiles. ResponseNet [68] formulates a minimum-cost flow optimisation problem that is solved by linear programming. RegMod [69] was proposed to find disease-associated modules using interactions and gene expression data; this approach uses the support vector regression method with a diffusion kernel in order to find active modules. stSVM smoothes a vector of t statistics by mean of a random walk kernel and uses a support vector machine (SVM) to select a set of significant genes [77].

In this paper, we describe a pipeline to outline network regions enriched in statistics derived from different types of omics datasets (Fig. 3.1). We show that the network smoothing index (S), a network diffusion-based quantity introduced here, is a simple and informative measure to jointly quantify the amount of "-omics" information associated with a molecular entity (e.g. gene, mRNA, protein) and the information in network proximity to it. Consequently, we describe two general applications of S for finding differentially enriched regions, in relation to the type of input statistics S is derived from: the variation of S between two sets of samples (ΔS) or the permutation-adjustment of S (Sp) for, respectively, descriptive statistics or inferential statistics. We also describe a procedure (network resampling) for the assessment of the presence of significantly connected components among entities with the highest ΔS or Sp.

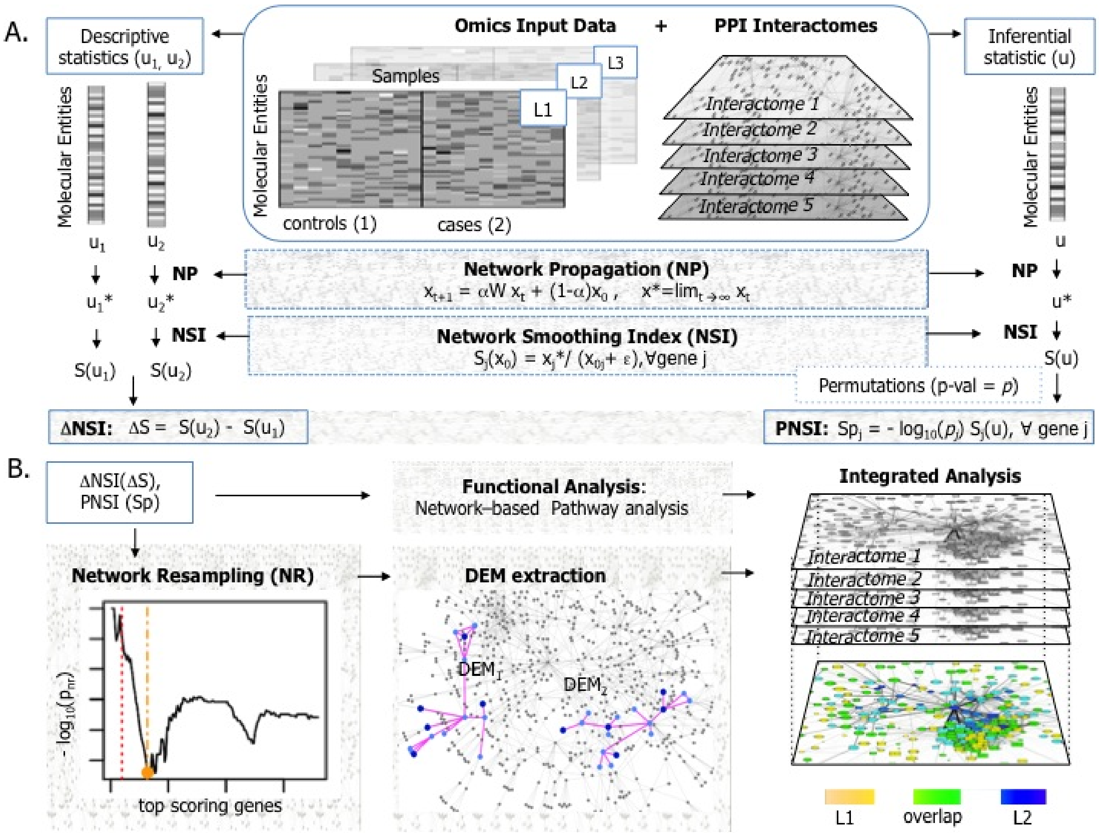

A.
Descriptive statistics (u_1, u_2)
Omics Input Data + PPI Interactomes
Inferential statistic (u)
Molecular Entities
Samples
L1
L2
L3
controls (1)
cases (2)
Interactome 1
Interactome 2
Interactome 3
Interactome 4
Interactome 5
u_1
u_2
u
NP
Network Propagation (NP)
$x_{t+1} = \alpha W x_t + (1-\alpha)x_0$, $x^* = \lim_{t\to\infty} x_t$
u_1^*
u_2^*
u^*
NSI
Network Smoothing Index (NSI)
$S_j(x_0) = x_j^*/(x_{0j} + \varepsilon), \forall$gene j
$S(u_1)$
$S(u_2)$
S(u)
Permutations (p-val = p)
ΔNSI: $\Delta S = S(u_2) - S(u_1)$
PNSI: $Sp_j = -\log_{10}(p_j) S_j(u), \forall$ gene j
B.
ΔNSI(ΔS), PNSI (Sp)
Functional Analysis: Network-based Pathway analysis
Integrated Analysis
Network Resampling (NR)
DEM extraction
$-\log_{10}(p_{nr})$
top scoring genes
DEM_1
DEM_2
L1
overlap
L2

Figure 3.1: **Network-diffusion based analysis of omics for the identification of differentially enriched network regions.** **(a)** Statistics (descriptive on the left, inferential on the right) carrying molecular information are smoothed by means of network propagation, and the NSI score is computed. **(b)** Identification of significantly connected components among genes ranked by ΔS or Sp, and network-based functional characterisation.

We show the performance of network diffusion, ΔS, Sp and network resampling in a simulated dataset. Then, as a proof of principle, we apply these tools to spot PPI network regions differentially enriched in somatic mutations (SM) and gene expression (GE) variations between two prognostic groups of patients affected by prostate adenocarcinoma (PRAD). We carry out the analyses of molecular profiles using five datasets of molecular interactions.

The strategy described here can be in principle applied to two-classes analyses of any high-throughput dataset that can be mapped to a network of interactions. We implemented the pipeline used in our study into an R package available upon request for non-commercial entities.

3.1.1 Identification of differentially enriched modules

Network diffusion methods can be applied to different types of initial quantities, like molecular entities-by-samples matrices [46] and real valued summary statistics [66, 67, 77]. Such differences are mainly motivated by the type of input data, the objective of the analysis and the algorithm used to generate the results. We consider two apply network diffusion on two types of input: descriptive statistics that summarise the information of a group of samples (Fig. 3.1**a** left-hand side); inferential statistics that describes the molecular variations between two classes (Fig. 3.1**a** right-hand side).

In relation to the physical model of diffusion we have used, we refer to the positive elements of the input as "sources" and to the represented molecular quantity as information or fluid. Network diffusion allows to "smooth" the information associated with molecular entities according to a given pattern of interactions among entities, encoded in the adjacency matrix $\mathbf{A}$, a square binary matrix where positive elements a_{ij} indicate the presence of an interaction between entities i and j. We consider the diffusion method designated as "network propagation" [57] to smooth any input statistic x_0 :

$$x_{t+1} = \alpha \mathbf{W} \cdot x_t + (1 - \alpha) x_0 \tag{3.1}$$

where $\mathbf{W}$ is a symmetrically normalised version of $\mathbf{A}$ (see Methods) and $0 < \alpha < 1$ controls the contribution of the two addends. At each iteration t, the amount of information in each vertex is the sum of its initial information and the total amount of information associated with its neighbours at the previous iteration. This iterative procedure will converge in a finite number of iterations to a particular state x^* [78]. Note that we can interpret the iterative procedure of equation (3.1) as a diffusion process in which a fluid enters from sources, flows through the links between vertices and exits at a constant first order rate from each vertex. In particular, after a proper rescaling, network propagation is equivalent to the laplacian dynamics of the open system of type $dx/dt = -L'x + b$, where $L' = \alpha W - I$ and b represents the molecular profile; this equivalence implies that the steady state reached by the laplacian dynamics is the same state x^* to which equation (3.1) converges (see Supplementary Note S1 online). At steady state, high values are associated with sources and with vertices in network proximity to sources. Note that network diffusion, in contrast to other methods, is a global measure of network proximity, i.e. it considers the whole network [44].

In order to quantify the average amount of information at steady state (x^*) in relation to the initial one (x_0) in a subset of samples, we introduce the network smoothing index (NSI) S_j of a molecular entity j:

$$S_j(x_0) = \frac{x_j^*}{x_{0j} + \epsilon} \tag{3.2}$$

where ϵ is a parameter that weights the relative importance of initial and final states. Small values of ϵ underline the gain of information in relation to the initial state, while when $\epsilon \to \infty$ only the final state (x_j^*) matters. A reasonable compromise can be found in order to prioritise both sources and entities in network proximity to sources (see below the results for PRAD data).

At this point, NSI based on within-class statistics relative to a set of controls (u_1) can be subtracted to the NSI calculated on within-class statistics relative to a set of cases (u_2):

$$\Delta S_j = S_j(u_2) - S_j(u_1) \tag{3.3}$$

where the ΔS_j jointly quantifies the differential amount of molecular variation observed in entity j and in its neighbourhood between two classes of samples. Note that the calculation of ΔS_j contrasts the effect of hubs that assume high S_j in both classes only because of their centrality. In other words, since the topology of the network is the same for the two subsets of samples, the effects ascribable only to topology are mitigated.

If the NSI is obtained from the smoothing of a differential statistics (u) then permutations can be used to mitigate the effect of hubs. In this case, we define the PNSI (Sp) value for each gene j:

$$Sp_j(u) = -log_{10}(p_j) \cdot S_j(u) \tag{3.4}$$

where p_j is the fraction of times an S_j obtained from the smoothing of a randomised differential statistic is equal or greater than the real S_j. The quantities $S, \Delta S$ and Sp are vectors of length equal to the entire number of molecular entities considered as well as the input within-classes or differential statistics.

At this point, the top molecular entities sorted in decreasing order of ΔS (or Sp) belong to regions with a differential content of information. In order to identify one or more differentially enriched modules we need to cut this list and extract the subnetworks composed of such top entities. Accordingly, we define the non-decreasing objective function Ω for ΔS (or Sp):

$$\Omega(n) = \Delta S^T(n) \cdot \mathbf{A}_n \cdot \Delta S(n) \tag{3.5}$$

where $\mathbf{A}_n$ is the adjacency matrix for only the first n top scoring molecular entities. In other words, the function $\Omega(n)$ is the sum of all the products $\Delta S_i \Delta S_j$ between the pairs (i, j) of interacting ($a_{ij} = 1$) molecular entities. According to the local hypothesis [6], if the difference between the two classes is the consequence of an underlying biological function or pathobiological process, we should expect a significant pattern of connection among the molecular entities with the highest ΔS [65]. In order to quantify such significance, for each rank n, we calculate the values of $\Omega(n)$ using k resampled adjacency matrices, where we randomly assign the existing links among vertices conserving the same degree distribution. Then, we calculate the corresponding network resampling p values (p_{nr}), which are equal to the mean number of times a random assignment of the links among the first n molecular entities determines a value of $\Omega(n)$ higher than or equal to the one observed with real links (see Supplementary Note S2 online). Following this procedure, the ranks associated with low values of p_{nr} indicate the presence of connected genes with high ΔS (or Sp).

3.1.2 Performance on simulated data

We have designed a series of simulated datasets to study the NSI ability to prioritise genes belonging to network regions (shortly modules) with a higher content of omics information in comparison to the rest of the network. We have considered the generic definition of module as random subnetworks, where the existence of a finite path that connect each pair of module gene is the only topological requirement, because disease proteins do not necessarily reside within locally dense communities [65] and, more generally, it is not clear to which extent functional modules, topological modules and disease modules overlap [6]. We have associated with each module a specific amount of signal (ω) non-uniformly distributed among the genes, in order to have a few module genes contributing to the most of the signal and all the other module genes with lower or not significant amounts of signal (Fig. 3.2**a**). This distribution was inspired by what is observed in real datasets, like the “mountains” (highly mutated genes) and “hills” (genes altered infrequently) observed in cancer mutation landscape [79]. Moreover, it models a more general scenario in which the alteration of some module genes is observed in many individuals (higher signal), other module genes are altered more specifically (lower signal) and, lastly, some module genes are marginally altered. Conversely, outside the module the signal was randomly distributed. The simulated datasets were defined such that the real amounts of mutation per patient and per gene were not modified.

We have explored several configurations, varying ω, the distribution of ω, the parameter ϵ (equation 3.2), module size and module topological density. We used STRING [80] and PRAD SM data (from TCGA [81]) as sources, respectively, of molecular interactions and biological signal (see methods). For each confuguration we have computed S and calculated the recall as the fraction of module genes that appear among the top M genes ranked in descending order of S, where M is module size.

We have observed high recalls either in modules enriched in mountains and in those enriched in hills (Fig. 3.2**b**). When the biological signal is particularly high the best performance is obtained for high values of ϵ, while when the module is composed of a mixture of genes with strong and marginal variation , we have observed the maximum recall for smaller values of ϵ ($\epsilon \approx 0.25$) (Fig. 3.2**c**). The performance of the NSI increases with increasing topological density (number of existing links over all possible links among the module's genes (Fig. 3.2**d**)). This behaviour is particularly highlighted for low values of f_h, underlying that a high density of connections strengthens the ability of the index to prioritise genes in network proximity to those with a high content of molecular alterations. The use of S determines higher recall than the non-network quantity f (variation of relative frequency of gene mutation), apart from the extreme case in which the module genes are exactly the top ranking genes by f (Fig. 3.2**e**). S determined the identification of more connected network regions with a higher content of module genes compared to what we observed using f (Fig. 3.2**f**).

We assessed the ability of network resampling in the prediction of module size. For this analysis, we have ordered genes by ΔS, which quantified the difference between a simulated dataset with a gene module enriched in biological information (as described above) and a simulated dataset without such enrichment (see methods). Also in this case, the real amounts of mutation per patient and per gene were not modified. As the signal ω increases, the size of significantly connected components approaches module's size (Fig. 3.3). We observed that the optimisation of p_{nr} values reaches a good accuracy around $\omega = 12\%$. The difference that we have observed, for lower values of ω, between the size of significantly connected components and module's size M, indicates that differential information (amount of mutations) is enriched in a subregion of the gene module (Fig. 3.3).

3.1.3 Prostate adenocarcinoma

As a proof of principle, we have applied the network-based pipeline to the identification of molecular interaction networks enriched in genes with a higher content of SMs and GE differences between two distinct PRAD prognostic grade groups, G5 and G2, where the higher the grade the poorer the prognosis. We have used these datasets to illustrate two possible types of input data. In particular, in the case of SM data, we have calculated the relative frequency of gene mutation within

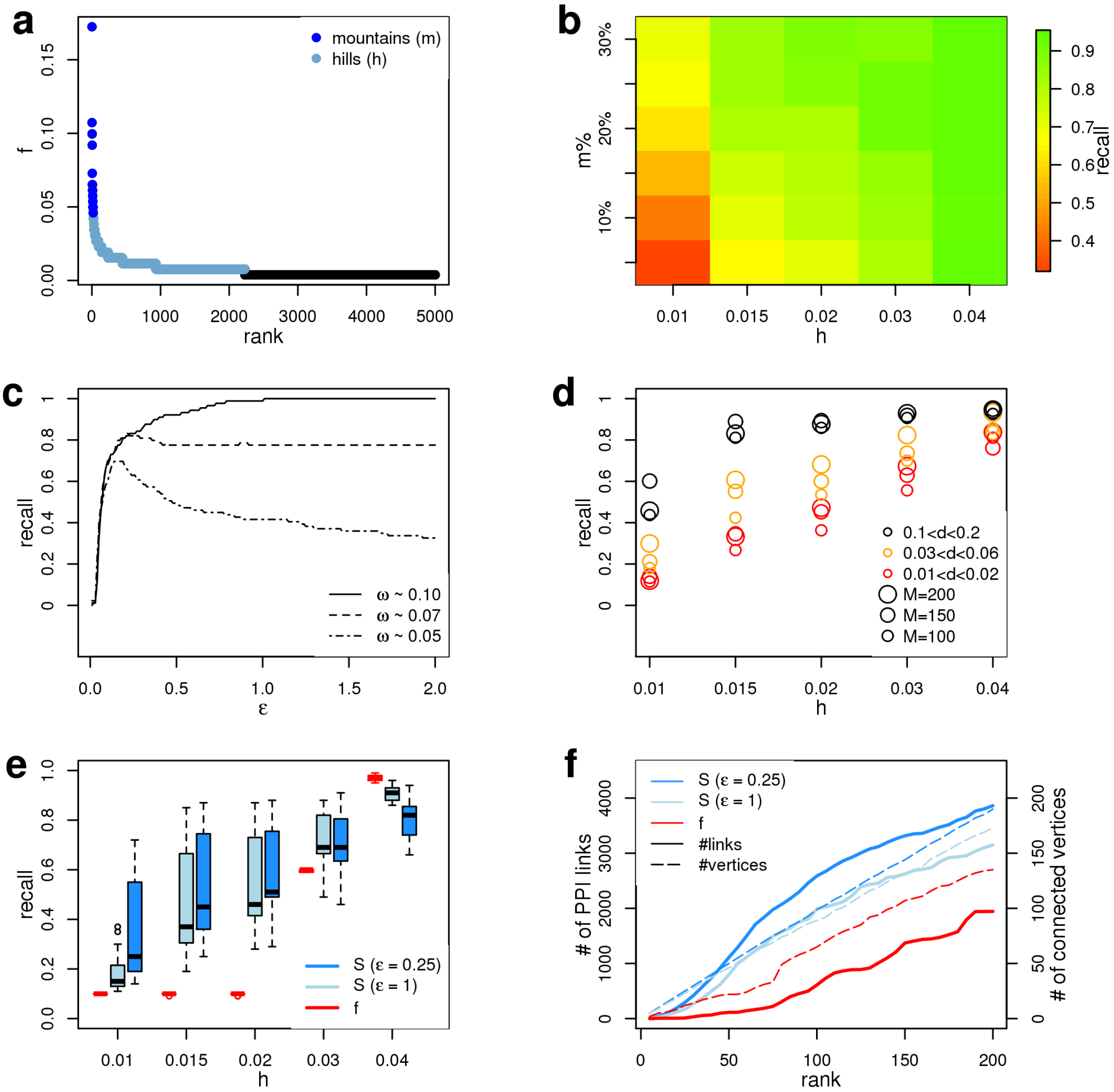


Figure 3.2: **Performance of differential network smoothing index in simulated datasets containing gene modules enriched in omics information.** **(a)** Somatic gene mutation relative frequencies ranked in decreasing order to underline mountains and hills. **(b)** On a sample of 100 vertices modules the recall heatmap with varying percentage of mountain genes ($m\%$) and average hill frequency (h). **(c)** The fraction of recalled genes for different values of parameter ϵ, on a typical 100 vertices module, with varying signal strength ($m_{\%} = 0.1$, the signal ω increases with h). **(d)** Average recall vs signal strength obtained on several toy datasets of different sizes (100, 150, 200 nodes) and topological density (d). **(e)** Comparison between the network smoothing index recalls ($\epsilon = 0.25$ and $\epsilon = 1$) and Δf recalls with varying signal strength obtained on a sample of several 100 nodes toy datasets. **(f)** the gain both in number of links (solid line) and number of connected vertices on top of f, and S. **(a-f)** Simulations were run using STRING PPIs.

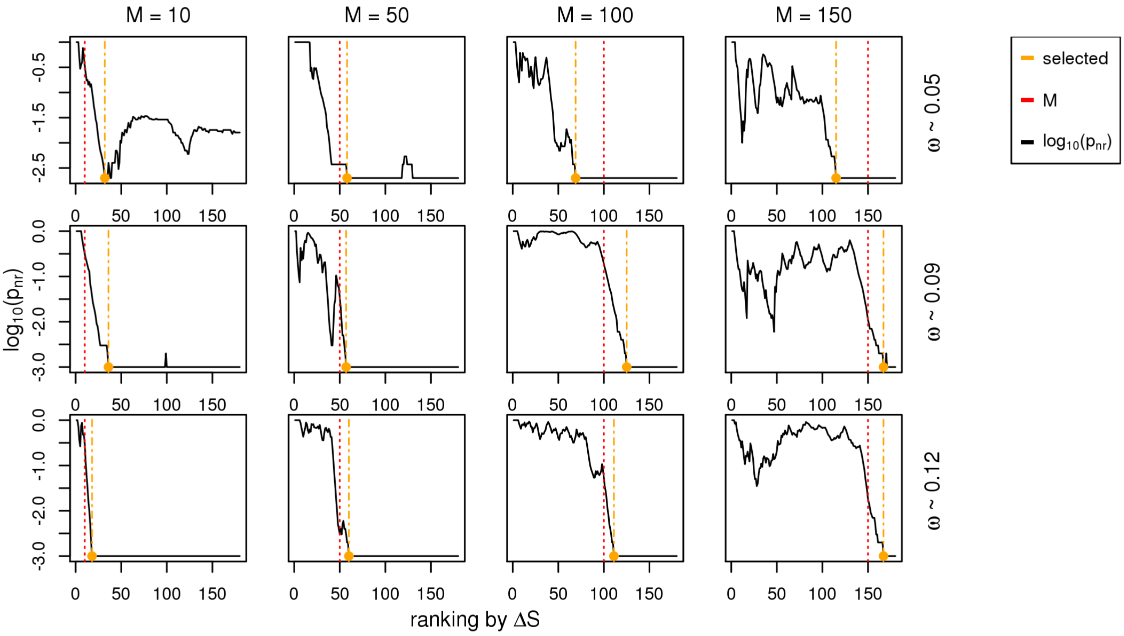


Figure 3.3: **Identification of significantly connected genes with network resampling p values in simulated datasets containing a gene module enriched in molecular alterations.** Network resampling p values (p_{nr}) calculated for each rank (n) of genes ordered by decreasing values of ΔS in datasets containing gene modules of different size (red lines, M) and signal (ω). Yellow lines indicate the smallest ranks associated with the presence of significantly connected components. Simulations were run using STRING PPIs.

each prognostic group (f), obtaining two vectors of descriptive statistics (f_1, f_2), and the variation between the two (Δf). We have applied network diffusion on f_1 and f_2 and calculated the corresponding S_1, S_2 and ΔS. In the case of GE data, we have calculated a differential statistics ($lfcp$, which combines basolute gene log fold change and adjusted p value of a moderated t statistics) between G5 and G2. We have applied network diffusion to $lfcp$, calculated the corresponding S and then Sp. We have repeated these analysis using five collections of direct (physical) and indirect (functional) PPIs (see Methods).

Of course, genes for which no interaction information is available in the considered interactome will not have a network-based value (Fig. 3.4 a-b). As expected, ΔS and Sp have prioritised genes jointly considering the relevance of the "network-free" statistics associated with each gene and the network-free statistics of genes in network proximity to each gene (Fig. 3.4 a-b). Genes with the highest variations of Δf or $lfcp$ are also associated with the highest values of ΔS or SP respectively. In particular, the overlap between network-free and network-based gene rankings can be tuned using the parameter ϵ (see [61] supplementary Fig. S2 and supplementary Fig. S3). Genes with similar values of Δf or $lfcp$ are discriminated in relation to their network location: the higher the network proximity of a gene to other genes associated with relevant Δf or lfcp, the higher its ΔS or Sp respectively. As a consequence, top ranking genes ordered by ΔS and Sp are more connected and form bigger networks than genes ordered by network-free quantities (Fig. 3.4 c-d).

We have applied the network resampling procedure to genes ranked by decreasing values of ΔS (enrichment of SM in G5 in comparison to G2) and Sp (enrichment in GE variations between G5 and G2), and found significantly connected modules in both cases (Fig. 3.5a). SM gene modules range from 109 and 231 genes depending on

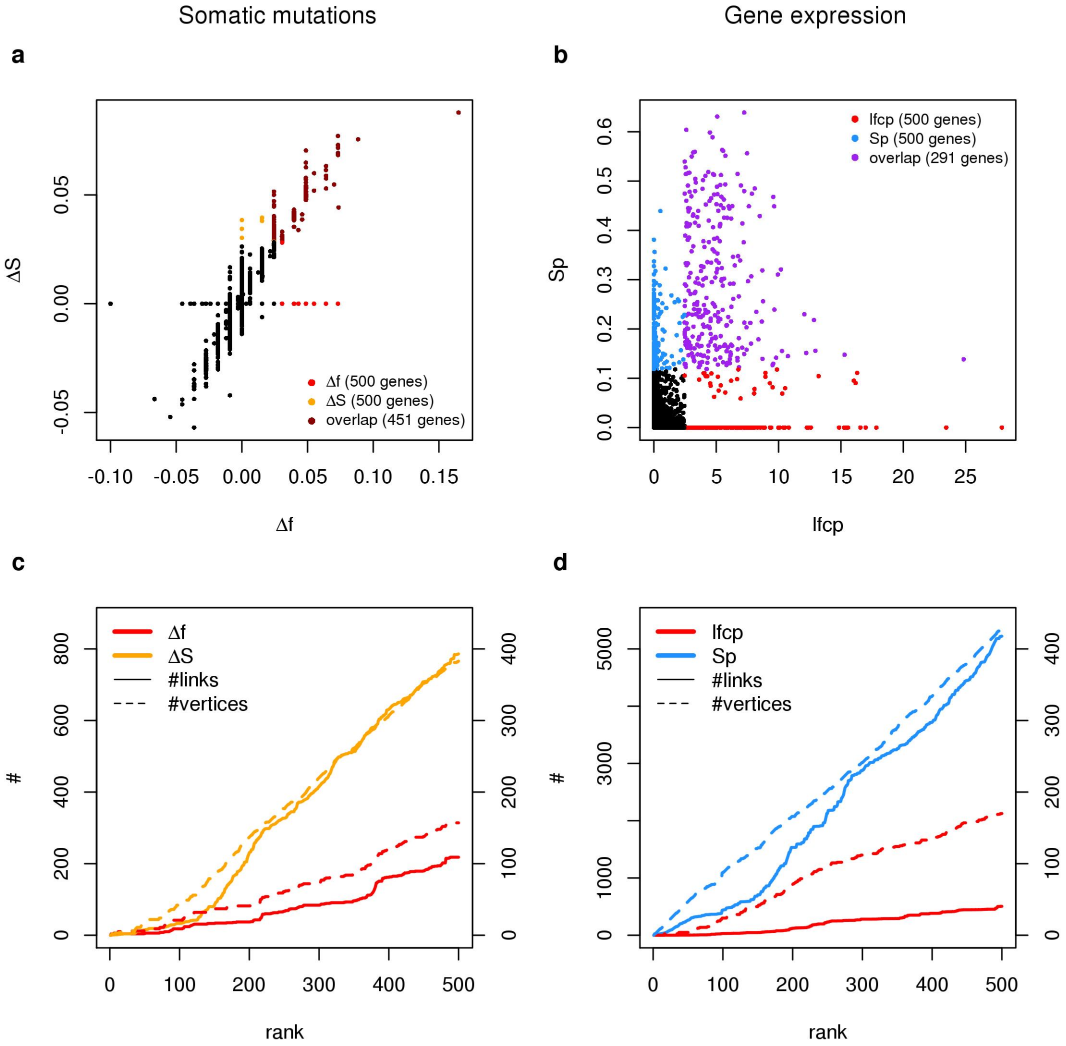


Figure 3.4: **Comparison of network-based and network-free quantities calculated on somatic mutation and gene expression data from PRAD samples associated with two different prognostic groups. (a-b)** Scatter plot with network ($y-$axis) *vs* network-free ($x-$axis) gene scores calculated on PRAD SM (a) and GE (b) data. **(c-d)** Number of links ($y-$axis, left) and number of vertices ($y-$axis, right) within the first 500 genes ordered by network (ΔS, Sp) and network-free (Δf, $lfcp$) gene scores calculated on PRAD SM (c) and PRAD GE (d) data. **(a-d)** ΔS and Sp were calculated using $\epsilon = 0.25$ and $\epsilon = 1$ respectively and using STRING PPIs. Legend: $\# =$ number of links (vertical axis, left) or number of veritces (vertical axis, right).

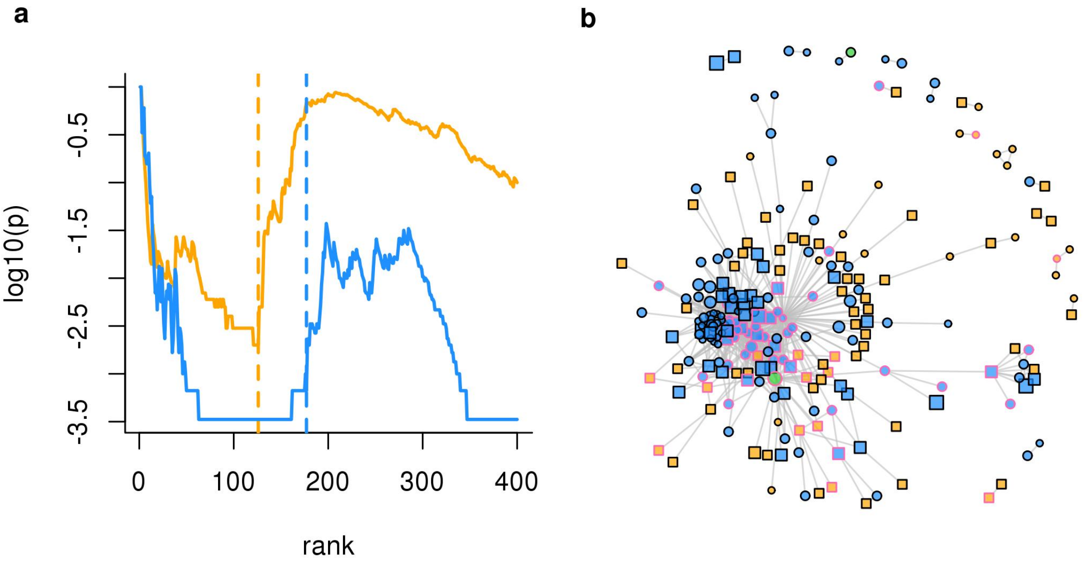


Figure 3.5: **Gene modules enriched in genes with different somatic mutations and gene expression levels between two PRAD prognostic groups.** **(a)** p_{nr} value of gene lists ranked by ΔS (SM, yellow) and Sp (GE, blue); vertical lines indicate the top ranking genes selected to be part of the corresponding gene modules. **(b)** Network of genes belonging to SM gene module (yellow), GE gene module (blue) or both (green); the pink border indicates genes that occur in at least 10 articles on PRAD. Legend: circle = gene found using networks; square = gene found using network and using network-free quantities (Δf or lfcp); vertex size = the larger the size the higher the gene score (maximum between ΔS and Sp).

the interactome (Tab. 3.1). A total of 342 distinct genes occur in these modules while 45 genes occur in all of them. Similarly, the GE gene modules range from 100 to 351 genes, with a total of 518 distinct genes and 33 found in all interactomes (Tab. 3.1). In addition to genes associated with the most extreme molecular variations between G5 and G2 (and therefore highly ranked also by network-free approaches) these modules contain genes specifically prioritised by ΔS and Sp. SM and GE modules contain genes that are highly cited in the literature of PRAD, some of which were specifically prioritised using networks (Tab. 3.2). The two genes TP53 and CDK2, the expression of which do not vary significantly, are examples of highly ranked genes because of their network proximity to differentially expressed genes (GE data), while, analogously, MEFV and TRPS1 are two examples of genes specifically found using networks in the analysis of SM data (Tab. 3.2). Other genes are not part of the current PRAD literature, but could be interesting candidate for further studies, since are in network proximity to genes with molecular alterations and/or already associated with the pathology ([61] supplementary Tabb. S1-2). Even if only a few genes belong to both SM module and GE module (e.g. TP53 and ANO4 using STRING, Supplementary Tabb. S5-6), several molecular interactions exist among genes of the two modules (Fig. 3.5b).

We carried out gene set enrichment analysis (GSEA) [82] to identify the molecular pathways regulated by genes with high ΔS and high Sp. We have found a total of 737 pathways with $p < 0.005$ (estimated with 1000 permutations) in at least one interactome, of which 270 in SM, 556 in GE and 89 in common. Comprehensively, the significant pathways cover the 8 capabilities (also known as hallmarks) acquired during the pathogenesis of cancer [83] (Fig. 3.6). The number of pathways found by GSEA with $p < 0.005$ (estimated with 1000 permutations) on gene lists gener-

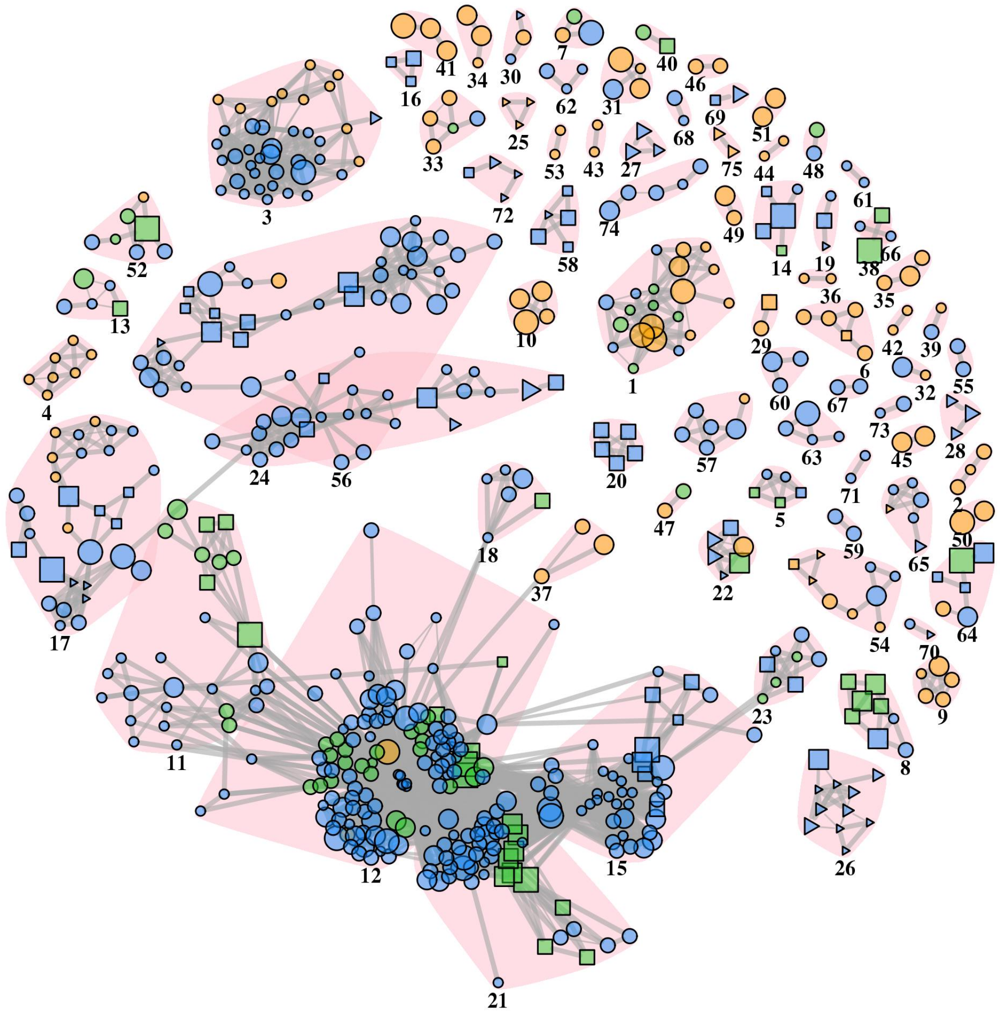


Figure 3.6: **Network of pathways enriched in genes with different somatic mutations and gene expression levels between two PRAD prognostic groups.** Vertices are pathways with $p < 0.003$ (GSEA, estimated with permutations) in at least one interactome and links indicate the similarity between pathways ($o \geq 0.95$); communities of similar pathways are underlined by pink background and numbers (Supplementary Tab. S8); pathways that are not similar to any other pathway are not shown. Legend: green = pathway found in SM and GE data; yellow = SM only; blue = GE only; circle = pathway found only when using network based quantities (ΔS or Sp); triangle = pathway found only when using network-free quantities (Δf or lfcp); square = pathway found by network-based quantities and network-free statistics; numbers refer to identifiers of communities of superpathways. (For complete legend see [61] supplementary Tab. S3).

Table 3.1: **Module size and common genes across interactomes**. Overview of the module size found in specific interactomes and the overlap among them.

Interactome (genes)	SM modules	GE modules
FP60	109	100
GHIASSIAN	117	351
HI	231	308
NCBI	117	100
STRING	126	177
$n \geq 1$	342	518
$n \geq 2$	132	257
$n \geq 3$	104	144
$n \geq 4$	77	84
$n = 5$	44	33

n: number of interactomes

Table 3.2: **Ranking of module genes with the highest occurrence in the literature of PRAD**. The occurrence is reported as number of papers; the symbol ”-” indicates genes not included in gene modules; these results are relative to STRING PPIs.

gene symbols	ΔS *(SM)*	Δf *(SM)*	*Sp (GE)*	*lfcp (GE)*	*citations*
TP53	1	1	155	2401	2075
PIK3CA	85	104.5	-	-	935
BIRC5	-	-	24	361	589
PTGS2	30	64	-	-	465
EZH2	-	-	57	412	395
CDK2	-	-	172	3324	391
CDK1	-	-	1	83	380
BRCA2	82	125.5	-	-	376
E2F1	-	-	44	235	305
CCNB1	-	-	174	639	284
CC-2	-	-	21	222	239
SERPINB5	-	-	67	151	239
CBX2	-	-	45	154	201
SMAD4	20	64	-	-	139
MEFV	73	707.5	-	-	81
HDAC6	42	64	-	-	48
CHD1	55	64	-	-	29
TRPS1	119	707.5	-	-	26
IDH1	14	16.5	-	-	14
MST1R	39	64	-	-	13

ated using networks are more than those found by GSEA on gene lists ordered by network-free statistics. Therefore, using networks it was possible to create a more comprehensive enrichment map, which displays pathways clustered in communities on the basis of common genes (Fig. 3.6). Apart few exceptions, the majority of pathways missed by network analysis are similar to pathways found by network analysis (Fig. 3.6).

3.1.4 Comparison with other diffusion-based methods

We have used a non parametric method (SAM [84]) to compare quantile normalised (QN), network-smoothed (NP) SM profiles of G5 and G2 (STRING PPIs), anal-

ogously to what was done in a recent work [46]. However, due to the sparsity of PRAD SM data, such approach (NP+QN+SAM) produced a gene ranking characterised by a small overlap with ΔS (and Δf, see Fig. 3.7a-b). In fact, many genes with a marginal difference of mutations between G5 and G2 were highly ranked by NP+QN+SAM, because these genes had very conserved differences of their quantile normalised, network-smoothed values between G5 and G2 (Fig. 3.7a-b).
We applied the stSVM method [77] on PRAD GE data (G5 and G2) and STRING PPIs. We have calculated S using the inferential statistics used by stSVM (t) and obtained a strong overlap among the top ranking genes ordered by NSI (Fig. 3.7c-e). An accurate description of the stSVM method is given in chapter 1.

3.2 Methods

3.2.1 Network Diffusion

The adjacency matrix $\mathbf{A}$ was normalised by dividing each element a_{ij} by the square root of the product of the degrees (k_i, k_j) of the corresponding vertices:

$$w_{ij} = \frac{a_{ij}}{\sqrt{k_i k_j}}$$

Network propagation (equation 3.1) was run iteratively for $t = [0, 1, 2, \ldots]$ until convergence: $|x_{t+1} - x_t| < 10^{-6}$. The choice of parameter α influences the behaviour of the diffusion algorithm, since α controls how much information is kept in vertices versus how much tends to be spread through the network. From a physical point of view it is reasonable to assume that $\alpha > 0.5$, which corresponds to an increase in the importance of network topology. Therefore, α was set to 0.7, a value that determined consistent results in previous studies [46, 76] and is a good trade off between diffusion rate and computational cost (which increases as $\alpha \to 1$).

3.2.2 Molecular interaction data

Five sources of PPI data were considered, abbreviated as STRING, NCBI, HI, FP60 and GHIASSIAN. Native identifiers were mapped to Entrez Gene [85] identifiers using NCBI data released June 26th 2015. STRING interactions were downloaded from STRING (version 10) web site, a database of direct and indirect PPIs [80]; in case multiple proteins mapped to the same gene identifier, only the pair of gene ids with the highest STRING confidence score was considered; a total of 11,535 genes and 207,157 links with confidence score $\geq$ 700 were retained. NCBI interactions were downloaded from NCBI ftp service, for a total of 15,098 genes and 159,092 links. HI protein links were collected from Rolland et al. [86] and a total of 7,760 genes and 25,040 links were obtained. FP60 interactions were collected from Kotlyar et al. [71] and a total of 10,363 genes and 258,923 links were retained. GHIASSIAN protein interactions were collected from Ghiassian et al. [65], for a total of 13,253 genes and 138,126 links.

3.2.3 Prostate adenocarcinoma data

PRAD clinical data were downloaded from the TCGA portal [81]. Prognostic grade groups based on the Gleason grading system were calculated as proposed

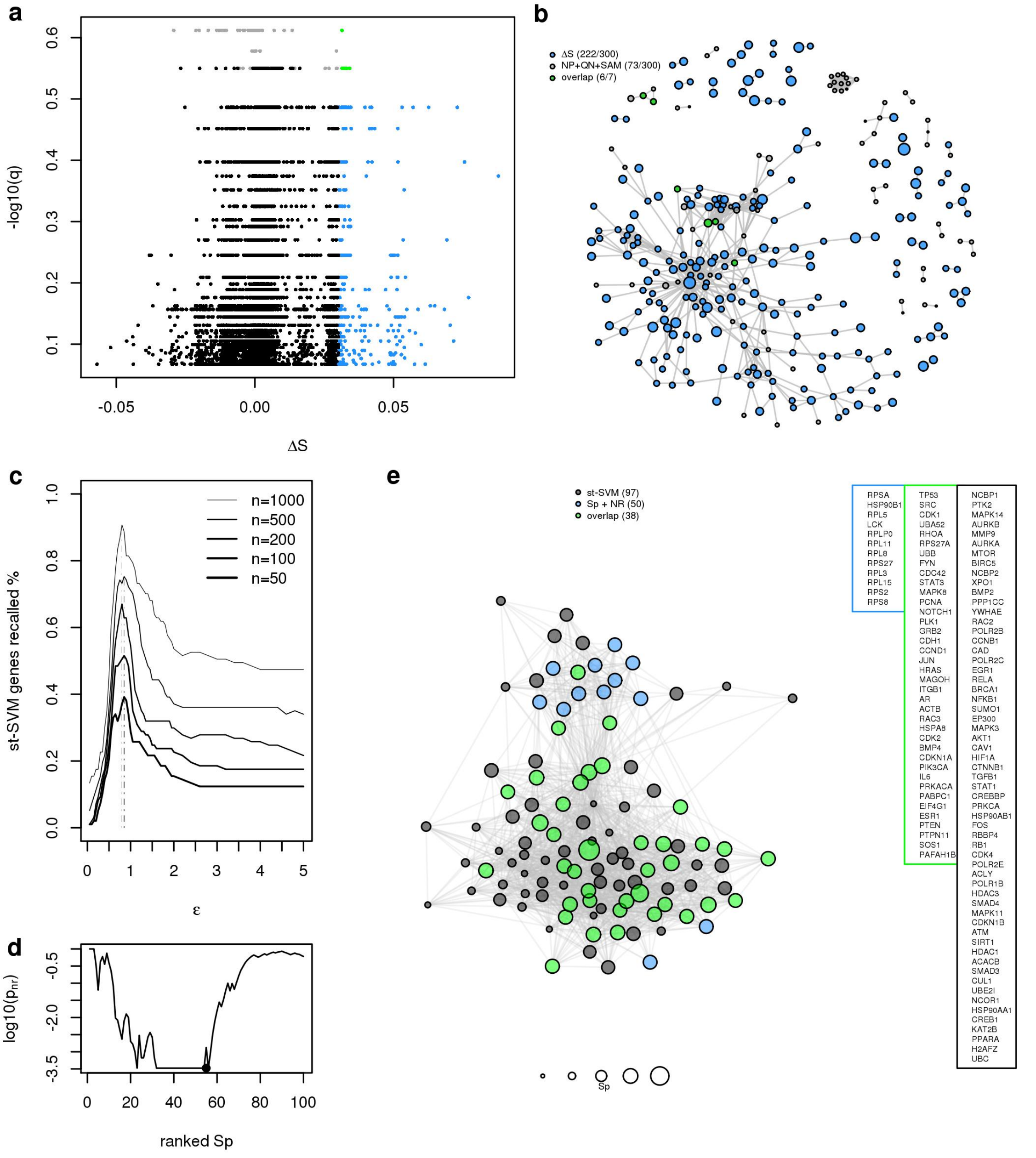


Figure 3.7: **Comparison of network smoothing index with other diffusion-based methods on two-classes analysis of PRAD somatic mutation (SM) and gene expression data (GE).** **(a)** Scatter plot with SAM q values and ΔS calculated on PRAD SM data. **(b)** Network of STRING PPIs formed by the top 300 genes order by SAM q or ΔS; the number of genes with at least one interaction is indicated between parenthesis. **(c)** Percentage of stSVM extracted genes recalled on top of the list (from 50 to 200) of Sp. **(d)** Network resampling applied to the Sp array suggests to cut around 55 top scoring genes. **(e)** Overlap between st-SVM and NSI on STRING interactome.

in Pierorazio et al. [87]: Gleason score ≤ 6 (prognostic grade group 1, G1); Gleason score 3+4=7 (G2); Gleason score 4+3=7 (G3); Gleason score 4+4=8 (G4); and Gleason score 9-10 (G5). Groups G2 and G5 contained respectively 110 and 41 subjects (see Supplementary Table S9 online).

Prostate adenocarcinoma (PRAD) curated somatic mutation (SM) data (collected with the Illumina Genome Analyzer platform) and PRAD RNA sequencing data (GE) (collected with the Illumina HiSeq 2000 RNA Sequencing (Version 2) platform) were downloaded from the TCGA portal for the 179 subjects. Only primary solid tumors (TCGA short letter code "TP") were considered. Both datasets were updated to Entrez Gene [85] identifiers released June 26th 2015.

SM dataset was composed of a total of 151 subjects (G2 and G5) with mutations in 6,898 genes (subjects with < 10 mutations or with > 200 were not considered). This dataset was encoded as a binary genes-by-samples matrix where the generic element a_{ij} was set to 1 if the patient j had at least one mutation in gene i, analogously to Hofree[46]. Then a vector of relative frequencies of gene mutation was calculated for each prognostic group.

Multiple gene expression profiles mapped to the same gene were collapsed considering the "MaxMean" criterion (implemented in the WGCNA package [88]). Only genes with more than 5 counts in at least 25% of subjects were considered. The dataset was normalized using the TMM method (trimmed mean of M values [89]) available in edgeR [90] R package , and log-cpm (count-per-milion) values were obtained using the "voom" function available in limma [91] R package. Only genes with cpm $>$ in at least 25% of subjects were considered. A total of 14,676 genes and 151 subjects (G2 and G5) were obtained. A vector of absolute differential statistics was calculated (as described in Xiao *et al.* [92]) from fold changes (FC) between G5 and G2, and the corresponding p values adjusted for false discovery rate (from limma [91]): $lfcp = -\log_{10}(p)|\log_2(FC)|$.

3.2.4 Simulated datasets

Simulated modules were defined as random subnetworks of the STRING [80] PPI network, as previously described in Mosca *et al.* [76]. Briefly, a "seed" gene is randomly selected and, then, up to 5 direct interactors are added to the current module. This procedure is repeated randomly selecting a new seed among the current module genes until the desired module size is reached. Note that this procedure defines connected subnetworks with different topological features (modularity, clustering coefficient, etc.).

The vector of frequencies of gene SM across individuals (f) was permuted, such that the initial sums of SM per patient and per gene across all subjects were not modified. At this point, genes labels were re-assigned in order to obtain the desired frequencies on the module. The re-assignment is controlled by the two parameters $m_{\%}$, the percentage of mountains (the highest frequencies) within the module, and h, the average mutation frequency of hills (genes with lower frequencies) (Fig. **??**a). We define the fraction of "signal" (ω) associated with a module as:

$$\omega = \frac{\sum_{j \in M} f_j}{\omega_{tot}} \tag{3.6}$$

where $\omega_{tot} = \sum_j f_j$. Therefore for any fixed value of $m_{\%}$, the amount of signal lying

on a given synthetic module increases with the average hill?s mutation frequency h and module size M (See Supplementary figure).
The recall was defined as the fraction of the top ranking genes sorted by decreasing order of ΔS that belong to the *a priori* considered module of size M: Recall $= \frac{|H \cap G|}{M}$, where H is the set of the first M genes ranked by decreasing order of ΔS and G are module genes.

3.2.5 Pathway analysis

Pathway analysis was carried out using the gene set enrichment analysis approach [82]. Genes were ranked in decreasing order of ΔS. NCBI Biosystems [93] was used as source of gene-pathway associations; only pathways with a number (n) of genes $10 \leq n \leq 300$ were considered. Enrichment scores and associated p values were calculated by means of the HTSAnalyzeR [94] R package using 999 permutations [95]. The p values calculated by HTSAnalyzeR were updated according to the equation $p' = (p \cdot 999 + 1)/1,000$, in order to count the real gene ranking as one among the 1,000 permutations, and then adjusted for false discovery rate using R function “p.adjust”. The similarity between two gene sets (A, B) was calculated using the overlap index: $o = \frac{|A \cap B|}{\min(|A|,|B|)}$.

3.2.6 Data mining of PRAD literature.

Literature-based text mining was performed using ProteinQuest (PQ) [96]. PQ is a web based platform for biomedical literature retrieval and analysis. PQ searches within PubMed abstracts and image captions from free full text articles. PQ text-mining tool parses target documents searching for terms related to curated ontologies (e.g. diseases, bioprocesses, pathways, body parts). Multiple searches for more than one alias were used to resolve ambiguities in the terminology. PQ was queried in order to retrieve the co-occurrence of genes and PRAD in the scientific literature.

3.2.7 Other diffusion-based methods.

Network smoothed somatic mutation profiles were quantile normalised with the normalizeQuantiles function of limma [91] R package. SAM statistics were computed with the samr [97] R package, using parameters “Two class unpaired” and “wilcoxon”. The netClass [36] R package was used as implementaton of stSVM [77].

3.3 Discussion and conclusions

We have introduced the network smoothing index (S), a network diffusion-based way of interpreting the molecular profiles in the context of an interaction network. S summarises the amount of omics information of an entity jointly with the amount of information of its network neighbourhood, defined considering the whole network topology via network diffusion. The comparison of S between two groups of samples (ΔS) is a network-based measure that indicates the differential amount of molecular variation and intrinsically mitigates the influence of topology on network smoothed values of the two groups. Alternatively, S can be adjusted by means of p values estimated with permutations, obtaining Sp.

In general, S, ΔS and Sp determine a network-based prioritization of molecular entities that highlights network regions enriched in molecular alterations. For example, such quantities allow: to find altered genes that are also involved in similar biological processes; to discriminate genes with similar molecular profiles, which is especially useful in case of ties; to highlight possible co-players of a pathological process, which have marginal molecular variations but are in network proximity to genes with relevant variations.

The complexity of biological networks makes the precise definition of a network region involved in a biological process or pathology a challenge, and several approximations or heuristics approaches exist to deal with this challenge [70]. We have shown that the application of network resampling to a list of genes sorted by ΔS or Sp suggests possible definitions of such regions on the basis of the significance of the distribution of ΔS or Sp values over the network.

Molecular entities sorted by ΔS or Sp values can be used as input for further analyses, including for example pathway analysis. When used in combination with a method of pathway analysis, like GSEA [82], ΔS or SP allow the quantification of molecular variations occurring in functional modules (pathway-topology based analysis [98]).

We have showed that network propagation, after proper rescaling, is equivalent to a physical model that describes the diffusion of a virtual quantity throughout a network [47, 48]. The connection of the two models allows a better understanding of the meaning of the used parameters and allows a better comparison with similar approaches.

As a proof of principle, we have calculated ΔS on somatic mutations and Sp on gene expression data from PRAD samples of different prognostic groups (G5 and G2). We have shown that ΔS and Sp highlight, respectively, network regions enriched in a higher content of SM and GE variations of G5 in comparison to G2. We have focused on $\Delta S > 0$, but also the opposite or its absolute value can be meaningful, depending on the objective of the analysis. A deeper investigation of PRAD biology is beyond the scope of our work, nevertheless, we provide several genes which are very likely to have a role in the different prognostic outcome. In fact, these genes lie in network proximity to genes already associated to PRAD and in regions of the PPI network enriched in mutated and/or differentially expressed genes. In line with the local hypothesis our analysis revealed the existence of a large connected component of genes that are associated with molecular variations (genetic mutations and/or differential expression) between subjects of different prognostic groups.

If the molecular variations are the consequence of an underlying biological function or pathobiological process and hence, in line with the local hypothesis, the molecular entities associated with such function or process are in network proximity, the network-based approach described in this work identifies a significantly connected component associated with the hypothetical biological function or pathobiological process.

Chapter 4

Perturbative approach

In this chapter we describe a novel network-based method for approaching the analysis of one or more layers of omics information. We call such method “perturbative” since the molecular alterations lying in an omic dataset are interpreted as a perturbation of the normal biological state. For example genes presenting somatic mutations or significant levels of over/under expression are considered as abnormalities that deviate the normal trajectories of information flow within the cell. The normal information flow between molecular species is represented by a random walk taking place on the network of physical interactions (2). In this work we limit ourselves to the PPI (protein-protein interaction network), but other important biological networks are available like metabolic networks, or gene regulation networks [99] to cite a couple. In this sense we map the molecular alterations on the network: a node carrying molecular alterations is defined as a query node (Fig. 4.1).

Up to this point there is no conceptual difference from the network propagation method described in the previous chapter. However instead of a direct application of the diffusion process in order to investigate biological insights (3), we focus on the differences between the stationary distributions $\vec{p_s}$ and $\vec{p}*$ of the random walk respectively without and with the perturbation. In this sense we aim to measure the perturbation in terms of how much it deviates the trajectories of information flow.

The networks considered for the analysis are mainly connected subnetworks of the PPI network that can either be pathways significant for a given disease or subnetworks enriched with molecular alterations. In order to mantain the probabilistic interpretation of the information flow on the network we connect the query nodes to an external node with which the query nodes exchange information, deviating from normal behavior (Fig. 4.1). In this context it is possible to define the perturbation as a matrix ΔL so that the perturbed dynamic of information flow on the network is driven by the matrix $L + \Delta L$ where L is the Laplacian matrix of the unperturbed network.

In the next sections we first describe the mathematical methods, then we study the applications of the method both on synthetic data and TCGA prostate cancer database (PRAD [81]).

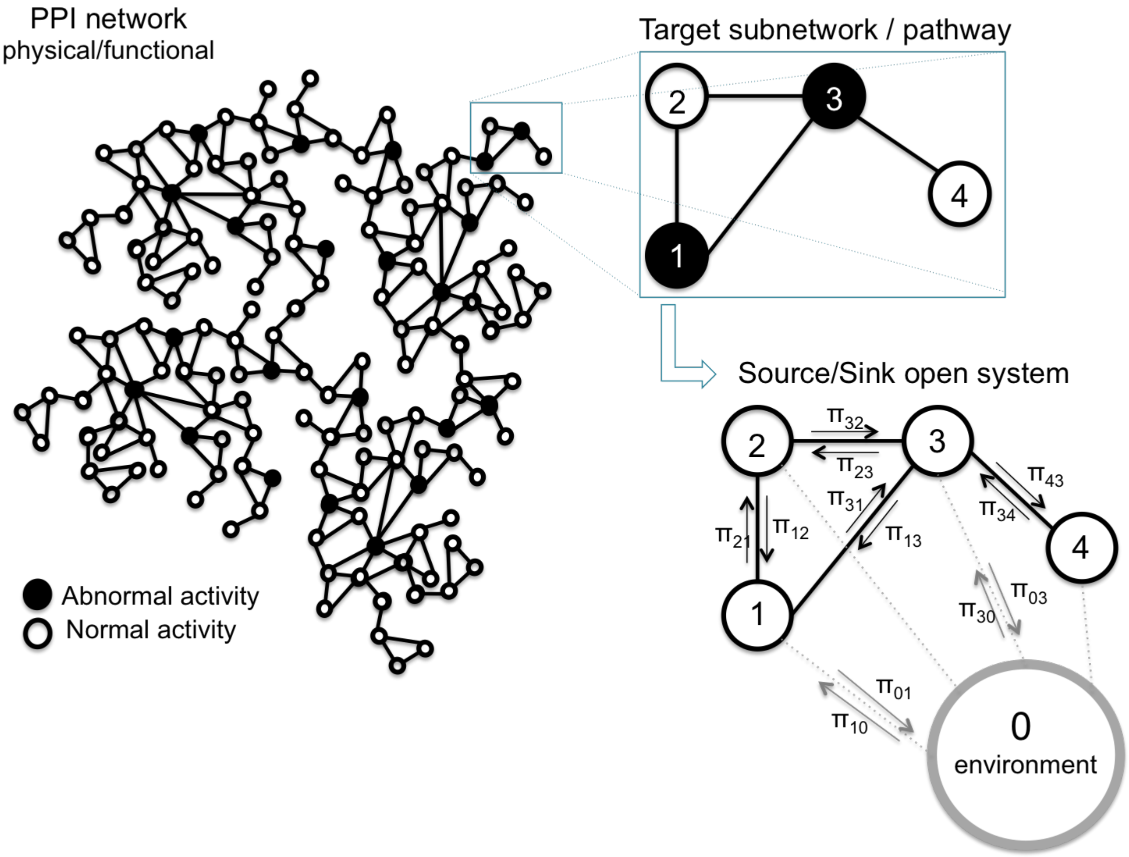


Figure 4.1: **Extended source/sink network.** From a given protein-protein interaction network and abnormal molecular information (e.g somatic mutations, SNP, over/under expressed genes) is extracted a smaller subnetwork (or pathway) to be analyzed. The introduction of the source/sink node (representing the environment) shifts the abnormal information from the query nodes of the network to the connections to/from the environment.

4.1 Methods: master equation external source

In chapter 2 we derived the general master equation model (2.33) for the random walk of N particles on an undirected network of M-nodes. Such model was built considering the dynamics of a finite number of particles N on the network and a regular condition (2.29) in the transitions between a node to another. We define π_{kj} the transition rate from node j to node k. Let $p_k(t)$ be the average probability to find a particle in the node k; we have the mean-field Master equation (2.68)

$$\dot{p}_k = \sum_j (\pi_{kj} p_j - \pi_{jk} p_k) \qquad k = 1, \cdots, M \tag{4.1}$$

that corresponds to the continuity equation for the constraint

$$\sum_k p_k(t) = 1$$

here we use the variable $\vec{p} = (p_1, p_2, \cdots, p_M)^T$ instead of $\vec{\phi}$ because of the probabilistic interpretation: the substance diffusing on the network $\vec{\phi}$ can be seen as the average probability for a particle to be found in a given node at a given time; it is convenient to introduce the Laplacian matrix

$$L_{kj} = d_k \delta_{kj} - \pi_{kj}, \qquad d_k = \sum_j \pi_{jk} \tag{4.2}$$

the master equation reads

$$\dot{\vec{p}} + L\vec{p} = 0 \tag{4.3}$$

and the stationary solution $\vec{p}_s = (p_1^s, p_2^s, \cdots, p_M^s)$ corresponds to the eigenvector of zero eigenvalue of the matrix L. If $d_k = 1$ then π_{kj} is a stochastic matrix. In a generic case L has all positive eigenvalues except the zero one, so that $\vec{p}_s$ is unique and attractive. Different methods can be applied to compute the stationary solution without solving the characteristic equation of the matrix L. We consider the problem of the presence of a source and sinks in the network. Let $p_0(t)$ the probability to introduce a particle in the source and π_{j0} the transition rate from the source to the node j, we have

$$\dot{p}_0 = s_0 - \sum_j \pi_{j0} p_0 \tag{4.4}$$

where s_0 is the source rate. Let $\epsilon = \sum_j \pi_{j0}$; we have the stationary state for the source

$$p_0^s = \frac{s_0}{\epsilon}$$

If we set $\epsilon = 1$ we preserve the stochastic character of the matrix π_{jk}. If we fix s_0 (or p_0), the stationary solution is determined. Let us define π_{0k} the transition probability to enter in the sink from the node k, the master equation is modified as

$$\dot{p}_k = \sum_{j \geq 0} (\pi_{kj} p_j - \pi_{jk} p_k) \tag{4.5}$$

The modified master equation has a M+1 components stationary solution $\vec{q}* = (p_0^s, \vec{p}*)^T$. With abuse of notation in the sequel we consider $\vec{p}*$ a M+1 components vector, therefore identifying $\vec{q}*$ with $\vec{p}*$. Indeed using the definition of p_0^s we get the non-homogeneous system.

$$\dot{p}_k = \sum_{j>0} (\pi_{kj} p_j - \pi_{jk} p_k) + \pi_{k0} s_0 - \pi_{0k} p_k \tag{4.6}$$

that can be solved since the matrix of the corresponding homogeneous system is invertible.

Lemma The stationary solution $\vec{p}* = (p_0^*, p_1^*, \cdots, p_M^*)^T$ satisfies the condition

$$s_0 = \sum_k \pi_{0k} p_k^* \tag{4.7}$$

Proof : from the equation

$$\sum_{k \geq 0} \dot{p}_k = \sum_j \pi_{j0} p_0 + \sum_k \sum_{j \geq 0} (\pi_{kj} p_j - \pi_{jk} p_k) = s_0 - \sum_k \pi_{0k} p_k$$

the condition (4.7) follows. □

Then if we consider the extended master equation (4.6) where we have substituted condition (4.7), the stationary solution of the extended system is formally equivalent to the solution of the isolated system. We remark that

$$\sum_{j \geq 0} p_j = const$$

so that the constraint (4.7) defines univocally the (non-normalized) stationary solution. We define the extended Laplacian matrix

$$L_{kj}^{ex} = d_k^{ex} \delta_{kj} - \pi_{kj}, \qquad d_k^{ex} = \sum_{j \geq 0} \pi_{jk}, \quad k, j \geq 0 \tag{4.8}$$

and the stationary solution of the extended system is the eigenvector of zero eigenvalue for L^{ex}. The structure of the extended Laplacian matrix is the following: the first row contains the transition rates which defines the dynamics of the source (i.e. the transition rates from the nodes to the source due to the dissipation); the first column defines the transition rate form the source to the nodes (i.e. the forcing terms). Finally the diagonal terms are modified with respect to the initial Laplacian matrix to preserve the Laplacian character of the extended matrix. It is convenient to write L^{ex} in a perturbative form $L^{ex} = L_0 + \Delta L$ where

L_0

$$\begin{pmatrix} 0 & 0 & 0 & \cdots & 0 \\ 0 & \pi_{11} & -\pi_{12} & \cdots & \pi_{1M} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & -\pi_{M1} & -\pi_{M2} & \cdots & \pi_{MM} \end{pmatrix}$$

$\Delta L =$

$$\begin{pmatrix} \epsilon & -\pi_{01} & -\pi_{02} & \cdots & -\pi_{0M} \\ -\pi_{10} & \pi_{01} & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ -\pi_{M0} & 0 & 0 & \cdots & \pi_{0M} \end{pmatrix}$$

Both the matrices are Laplacian. The constraint (4.7) points out the effect of the choice of s_0 on the stationary solution for fixed π_{0k}. Control theory should consider the effect of changing π_{0k} and π_{k0} on the stationary solution. It is interesting to compute the condition which preserves the stationary solution $\vec{p}_s$ of the isolated system (4.1). By a direct substitution we have:

$$\frac{\pi_{k0}}{\epsilon} - \pi_{0k} p_k^s = 0 \qquad \forall k$$

that is equivalent to a detailed balance condition between the source and each sink, since the probability flow is zero for each link.

Remark When we build the matrix L_0, the source must be independent from the rest of the network so that for any quantity s_0, the vector $(s_0, \vec{p}_s)^T$ is still a stationary solution of the extended system $\dot{\vec{q}} + L_0 \vec{q} = 0$.

4.1.1 Exact solution of the perturbed system

Introducing a source-sink perturbation to this isolated system we get an open system we can re-write equation (4.6) as

$$\begin{aligned} \dot{\vec{p}} + L \cdot \vec{p} - s_0 \vec{\pi}_{in} + I \cdot \vec{\pi}_{out} \cdot \vec{p} &= 0, \\ \dot{\vec{p}} + (L + I \cdot \vec{\pi}_{out}) \cdot \vec{p} - s_0 \vec{\pi}_{in} &= 0 \end{aligned} \tag{4.9}$$

where $\vec{p}$ has M components, s_0 is the source rate, $\vec{\pi}_{in} = (\pi_{10}, \cdots, \pi_{M0})^T$ are the weighted connections to the nodes from the source and $\vec{\pi}_{out} = (\pi_{01}, \cdots, \pi_{0M})^T$ are the weighted connections to the sink from node each node. The stationary solution of equation (4.9) is unique and depends on the source-sink choices; it is also easy to verify that it is formally equivalent to the the steady flow solution of the open system described in chapter 2 (2.70):

$$\vec{p}* = k(L + I \cdot \vec{\pi}_{out})^{-1} \cdot \vec{\pi}_{in} \tag{4.10}$$

with k being an appropriate constant. Assuming the probababilistic interpretation of the substance moving in the network and the addition of the source/sink node with the constraint (4.7) the solution of the perturbed system can be solved with equation (4.10) if and only if $k = s_0/\epsilon$, where $\epsilon = \sum_j \pi_{j0}$.

Proof We write an extended master equation by using an extended Laplacian matrix

$$\dot{\vec{q}} + L^{ex} \cdot \vec{q} = 0, \tag{4.11}$$

where now $\vec{q} = (p_0, \vec{p})^T$ has size $M + 1$ as well as the $(M + 1) \times (M + 1)$ extended Laplacian L^{ex}:

$$\begin{pmatrix} \epsilon & -\vec{\pi}_{out}^T \\ -\vec{\pi}_{in} & L + I \cdot \vec{\pi}_{out} \end{pmatrix}$$

By direct substitution we see that

$$L^{ex} \cdot \vec{q} = 0 \Leftrightarrow (a), (b)$$

where (a) and (b) are

$$\begin{array}{ll}(a) & p_0\epsilon - \vec{\pi}_{out}\vec{p} = 0\\ (b) & -\vec{\pi}_{in}p_0 + (L + I\vec{\pi}_{out})\vec{p} = 0\end{array}$$

in equation (a) the second addend is equivalent to s_0 if and only if condition (4.7) holds; so we can subtitute (a) $p_0 = s_0/\epsilon$ into (b) and find

$$\vec{p}* = \frac{s_0}{\epsilon}(L + I \cdot \vec{\pi}_{out})^{-1} \cdot \vec{\pi}_{in} \quad \Box$$

4.1.2 Iterative scheme set up

The effect of external source on the stationary solution of a master equation can be understood using a perturbative approach starting from the extended Laplacian matrix L_0 where the source s_0 is initially decoupled with the initial network. The stationary state is the direct product of the stationary state of the initial network $\vec{p}_s$ and any state p_0 of the source. The introduction of the connections π_{k0} can be seen as a perturbation ΔL of the matrix L_0 (to quantify the perturbation we could use a matrix norm). The problem is to compute the change in the stationary state $\vec{p}*$ due to the perturbation ΔL. We explicitly consider the case when the Laplacian matrix L is self-adjoint respect to a scalar product: given any couple of vectors $\vec{w}, \vec{u} \in \mathbb{R}^M$

$$\vec{w} \cdot L\vec{u} = L\vec{w} \cdot \vec{u} \tag{4.12}$$

where

$$\vec{w} \cdot \vec{u} = \sum_{ij} w_i h_{ij} u_j$$

for a metric matrix H. This is the case when $L = L_g H$ where L_g is the graph Laplacian (see Fig. 2.2), while the matrix H is a diagonal matrix with the inverse of the node degree along the diagonal ($H = G^{-1}$). In such a case if $\vec{v}_1$ and $\vec{v}_2$ are two eigenvectors of L with eigenvalues λ_1 and λ_2 respectively ($\lambda_1 \neq \lambda_2$), it follows

$$\vec{v}_1 \cdot \vec{v}_2 = 0$$

Proof:

$$\lambda_1 \vec{v}_1 \cdot \vec{v}_2 = L_g H \vec{v}_1 H \vec{v}_2 = H \vec{v}_1 L_g H \vec{v}_2 = \lambda_2 \vec{v}_2 \cdot \vec{v}_1$$

Then if $\lambda_1 \neq \lambda_2$ the thesis follows $\Box$.

In Chapter 2 we showed that finding a scalar product with respect to which the matrix L results self-adjoint implies that the system is in detailed balance, a correct assumption in a closed isolated system. In addition the introduction of such scalar product also allows to find a base $\{\vec{v}_1, \cdots, \vec{v}_M\}$ of orthogonal eigenvectors of L.

4.1.3 Perturbation without the source/sink

Let us consider the following problem: we have a perturbed Laplacian matrix $L+\Delta L$; is it possible to compute in a perturbative recursive way the stationary solution

$$(L + \Delta L)p* = 0 \tag{4.13}$$

assuming that there exists a unique stationary solution $p*$ (apart from a normalizing condition); we leave the vector notation for $\vec{p}_s$ and $\vec{p}*$ to simplify the writing. Let p_s be the stationary solution of the unperturbed Laplacian L, we look for a stationary solution of the form

$$p* = \hat{p} - \alpha p_s \tag{4.14}$$

where $p_s \cdot \hat{p} = 0$, i.e. p_s is orthogonal to $\hat{p}$. We are decomposing the solution into two orthogonal components. Since $p_s = v_1$ (the eigenvector of null eigenvalue in matrix L), we can think of $\hat{p}$ as the component of $p*$ in the orthogonal complement of the stationary solution of the unperturbed system. In other words $\hat{p} = \langle v_2, \cdots, v_m\rangle$; the following equation holds

$$L\hat{p} = -\Delta L\hat{p} + \alpha\Delta L p_s \tag{4.15}$$

which is equivalent to the system

$$\begin{aligned} L\hat{p} &= -(I-\Pi)\Delta L\hat{p} + \alpha(I-\Pi)\Delta L p_s \\ \alpha p_s \cdot \Delta L p_s &= p_s \cdot \Delta L\hat{p} \end{aligned} \tag{4.16}$$

where we introduce the projector Π on the one-dimensional kernel of L

$$\Pi v := \frac{p_s \cdot (p_s)^T}{\|p_s\|} v$$

If $p_s \cdot \Delta L p_s = 0$, system (4.16) reads

$$L\hat{p} = -\Delta L\hat{p} + \alpha\Delta L p_s \tag{4.17}$$

has a unique solution $p*$ for any value of α: i.e. we can set $\alpha = 1$ and after normalize the solution.
If $p_s \cdot \Delta L p_s \neq 0$ (generic case) then we define

$$\alpha = \frac{\hat{p} \cdot \Delta L p_s}{p_s \cdot \Delta L p_s} \tag{4.18}$$

and we consider the system

$$L\hat{p} \;=\; -(I-\Pi)\Delta L\hat{p} + \frac{\hat{p} \cdot \Delta L p_s}{p_s \cdot \Delta L p_s}(I-\Pi)\Delta L p_s \tag{4.19}$$

If $\hat{p}$ is a non-trivial solution of previous system, then the staionary solution of (4.13) is given by (4.14) with α defined in (4.18). We compute the solution of the system (4.19) using a fixed point principle (contraction principle). Let's consider the sequence

$$\hat{p}_0 = 0, \qquad \alpha = 1$$

and

$$\begin{aligned} L\hat{p}_n &= -(I-\Pi)\Delta L\hat{p}_{n-1} + \alpha_{n-1}(I-\Pi)\Delta L p_s \\ \alpha_n &= \frac{p_s \cdot \Delta L\hat{p}_n}{p_s \cdot \Delta L p_s} \end{aligned} \tag{4.20}$$

It is now convenient to expand $\hat{p}_n$ on the eigenvectors of the unperturbed matrix L:

$$\hat{p}_n = \sum_{k=1}^{M} c_{n,k} v_k, \quad \forall n$$

Remark $c_{n,1} = 0$ since it must hold that $\hat{p}_n \cdot p_s = \hat{p}_n \cdot v_1 = 0$.
By direct substitution first equation in (4.20) gives

$$\sum_k c_{n,k}\lambda_k = -\sum_k c_{n-1,k}(I-\Pi)\Delta L v_k + \alpha_{n-1}(I-\Pi)\Delta L v_1 \tag{4.21}$$

By projecting on the j-th eigenvector we obtain $\forall j \geq 2$

$$\begin{aligned} \lambda_j c_{n,j} &= -\sum_k c_{n-1,k}\Delta\mathcal{L}_{j,k} \quad + \quad \alpha_{n-1}\Delta\mathcal{L}_{j,1} \\ \alpha_n &= \frac{1}{\Delta\mathcal{L}_{1,1}}\sum_k c_{n,k}\Delta\mathcal{L}_{1,k} \end{aligned} \tag{4.22}$$

where

$$\Delta\mathcal{L}_{j,k} := v_j \cdot \Delta L v_k = v_j^T H \Delta L v_k \tag{4.23}$$

4.1.4 Perturbation with the source/sink

To consider the case with an external source we define the extended matrix $L^{ex} = L_0 + \Delta L$. Since the unperturbed network is disconnected, the matrix L_0 has a two dimensional kernel, we look for a stationary solution of the perturbed system in the form

$$p* = \hat{p} + \alpha p_s + e_0 \tag{4.24}$$

where e_0 is a vector with only the 0-component (i.e. the source) equal to 1. $\hat{p}$ is orthogonal to both p_s and e_0, therefore

$$\hat{p} = \sum_{k\geq 0} c_k v_k \tag{4.25}$$

with $c_0 = c_1 = 0$ where $v_0 = e_0$ and $v_1 = p_s$. We consider the recurrence:

$$\begin{aligned} L_0\hat{p}_n &= -(I-\Pi)\Delta L\hat{p}_{n-1} + \alpha_{n-1}(I-\Pi)\Delta L p_s + (I-\Pi)\Delta L e_0 \\ \alpha_n p_s \cdot \Delta L p_s &= -p_s \cdot \Delta L\hat{p} - p_s \cdot \Delta L e_0 \\ \alpha_n e_0 \cdot \Delta L p_s &= -e_0 \cdot \Delta L\hat{p} - e_0 \cdot \Delta L e_0 \end{aligned} \tag{4.26}$$

where Π is the projection on the 2d kernel of L_0. The last two equations has to be linearly dependent, so one equation has to be sufficient. In the case of the existence of an orthogonal base for L, the projector Π can be explicitly computed using the normalized eigenvectors v_k. Let

$$\hat{p}_n = \sum_{k\geq 1} c_{n,k} v_k \tag{4.27}$$

where $v_1 = p_s$. Substituting decomposition (4.27) in system (4.26) we obtain

$$\begin{aligned}\sum_k c_{n,k} L_0 v_k &= -(I-\Pi)\sum_k c_{n-1,k}\Delta L v_k + \alpha_{n-1}(I-\Pi)\Delta L p_s + (I-\Pi)\Delta L e_0 \\ \sum_k c_{n,k}\lambda_k &= -(I-\Pi)\sum_k c_{n-1,k}\Delta L v_k + \alpha_{n-1}(I-\Pi)\Delta L p_s + (I-\Pi)\Delta L e_0\end{aligned}$$

the iterative scheme is defined by the following couple of equations, projecting equation (4.28) on the eigenspaces of v_j, and using definitions (4.23)

$$c_{n,j} = -\sum_{k>1}\frac{c_{n-1,k}}{\lambda_j}\Delta\mathcal{L}_{j,k} - \alpha_{n-1}\Delta\mathcal{L}_{j,1} - \Delta\mathcal{L}_{j,0} \tag{4.28}$$

$$\alpha_n = -\sum_{k>1} c_{n,k}\frac{\Delta\mathcal{L}_{1,k}}{\Delta\mathcal{L}_{1,1}} - \frac{\Delta\mathcal{L}_{1,0}}{\Delta\mathcal{L}_{0,0}} \tag{4.29}$$

When the sequence converges with the initial condition

$$\hat{p}_0 = 0, \qquad \alpha_0 = 1$$

we get the stationary solution of the extended system.

4.2 Numerical considerations and first results

So far we described an external source perturbation approach to a close isolated Laplacian system in which the perturbation takes the matrix form ΔL. We defined a numerical scheme that, startig from the stationary solution of the closed isolated system should converge to the stationary solution of the perturbed system. We demonstrated that the solution of the perturbed system always exists. However the behavior of the numerical scheme depends on the features of the perturbation matrix ΔL. In fact, given a perturbation matrix ΔL (fixing the "topology" of the perturbation) the iterative scheme converges or diverges depending on the values of the input s_0: if the input value is strong the dynamics of on the network may not be able to converge, since the perturbation implies a network dynamics that is sensibly different than the one observed in the closed system. On the other hand, fixing the input value s_0 there are topological configurations that may be much more perturbative than others. The main goal of the iterative scheme is not the approximation of the perturbed stationary solution (which anyways may be useful for big networks), but the definition of an intrinsic measure of network stability: fixing the appropriate boundary conditions there exist a critical perturbation value ΔL that distinguishes between "weak perturbations" (the iterative scheme converges) and "strong perturbations" (the iterative scheme diverges).

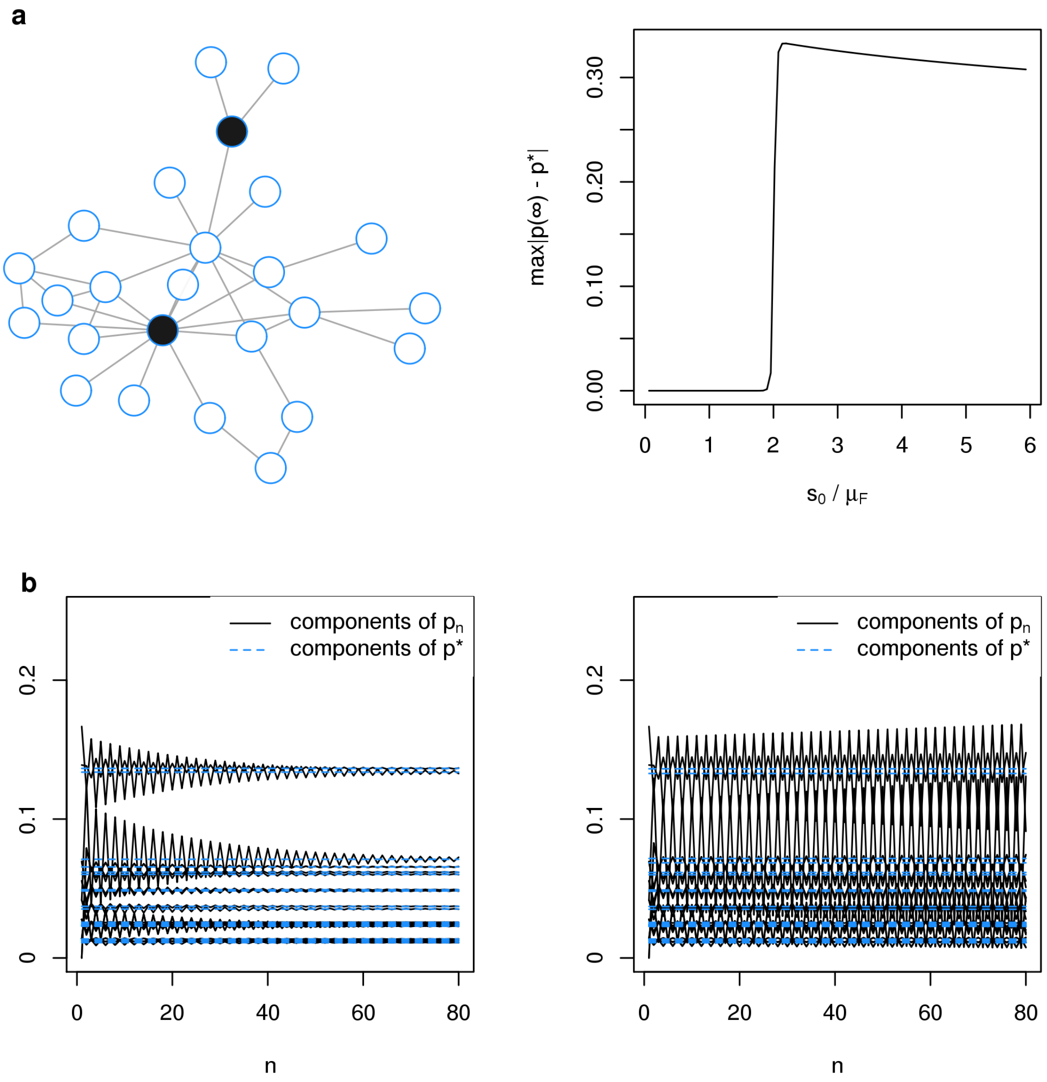


Figure 4.2: **Critical threshold. a.** We perturb a fully connected 25-nodes network with the source/sink node E connecting to the query nodes colored in black. On the right we plot the L1 distance between the exact stationary solution of the perturbed network and the correspondent quantity computed by the iterative scheme for increasing values of s_0. The jump corresponds to the critical threshold μ_t. **b.** We plot the components of the probability distribution evolution according to the iterative scheme before (left) and after (right) the critical threshold.

The critical threshold from a numerical point of view arises in equation (4.29) when we find positive values of the perturbation matrix $\Delta\mathcal{L}_{k,h}$ that divided by small eigenvalues start to contribute more to the coefficients $c_{n,k}$ of the eigenvectors v_k with $k > 1$ than to the coefficient α_n relative to the eigenvector $v_1 = p_s$. In particular the driving factor is the increase of $\Delta\mathcal{L}_{2,h}/\lambda_F$ where $\lambda_F = \lambda_2$ is the smallest eigenvalue after the null one, commonly called the Fiedler number of the graph. We can therefore say that from a spectral point of view the iterative scheme captures the first input value ΔL in which the stationary solution of the perturbed system $p*$ decomposed in the base of eigenvectors of L has a major contribution from the eigenvectors different from $v_1 = p_s$; technically this happens when the coefficients $c_{n,k}$ (in particular $c_{n,2}$) start to dominate the coefficient α_n.
In Fig. 4.2 on a synthetic dataset we a fully connected 25-nodes network with the source/sink node E connecting to the query nodes colored in black. For simplicity we assume the sink-rates equal to one another $\pi_{0i} = 1$ and we compute the distance between the exact stationary perturbed solution $p*$ and the approximation computed by the iterative scheme. We find the critical threshold μ_t where the iterative scheme starts to fail the approximation of the exact solution (Fig. 4.2a). The critical threshold at level of components consists of a critical value μ_t so that for $s_0 < \mu_t$ the components $\hat{p}_n$ converge to the exact perturbed solution $p*$, while for $s_0 > \mu_t$ the components of $\hat{p}_n$ present a divergent behavior (Fig. 4.2b).
The algorithm (4.29) can therefore be used to define an intrinsic measure of stability of a given network. When the perturbation ΔL is fixed from a topological point of view, the critical threshold consist of a critical input value s_0; when we vary the perturbation's topology (the number of query nodes and their positions) the concept of critical stability threshold is not a trivial definition. However we can find a lower boundary for critical stability threshold that is represented by the Fiedler number of the network λ_F

Lemma Given any connected undirected network with Laplacian matrix L and a perturbation ΔL the critical threshold is bounded by the Fiedler number: $\mu_t \geq \lambda_F$.

Proof : we consider the matrix norm $\|\cdot\|$ defined by:

$$\|M\| := \sup_v \frac{|Mv|}{|v|}$$

so that when M is symmetric its norm corresponds to the absolute value of its biggest eigenvalue. We operate a direct substitution in equation (4.29) and obtain:

$$c_{n,j} = \frac{1}{\lambda_j} \sum_{k=2}^{M} \left[\Delta\mathcal{L}_{jk} - \frac{\Delta\mathcal{L}_{j1}}{\Delta\mathcal{L}_{1,1}} \Delta\mathcal{L}_{1k} \right] c_{n-1,j}$$

We now consider the matrix $\Delta L'$ defined as:

$$\Delta L'_{ij} := \Delta\mathcal{L}_{ij} - \frac{\Delta\mathcal{L}_{i1}}{\Delta\mathcal{L}_{1,1}} \Delta\mathcal{L}_{1j}$$

The iterative scheme recurrence is driven by $\Delta L'$ and the inverse of matrix L in the subspace Π_0 so that the condition for (4.29) to be a contraction becomes:

$$\|L^{-1}\| \|\Delta L'\| < 1$$

using the fiedler number $\lambda_F = \lambda_2$ the previous contraction condition reads:

$$\frac{\|\Delta L'\|}{\lambda_F} < 1$$

We now underline that $\|\Delta L'\| \leq \|\Delta L\|$ since the perturbation $\Delta L'$ is the restriction of ΔL on the subspace Π_0. So we find a sufficient condition for the convergence of the recurrence:

$$\|\Delta L\| < \lambda_F \tag{4.30}$$

We can therefore affirm that the set of critical threshols has λ_F as a lower bound. □

Remark Because of the previous lemma it's appropriate to measure the critical threshold a perturbation ΔL in terms of ratio between critical input value μ_t and Fiedler number λ_F as well as the intensity of the input (s_0/λ_F).

Remark Condition (4.30) is an sense also a necessary condition for the convergence: if $\|\Delta L\| < \lambda_F$ it is always possible to find a perturbation matrix ΔL that degenerates the network dynamics.
When the sequence $\hat{p}_n$ diverges, the map (4.29) fails to be a contraction and one cannot follow by continuity the transition between the isolated and the perturbed stationary states. In the divergent case the perturbation is able to degenerate the zero eigenvalue and collapse the eigenvectors relative to $\lambda_1 = 0$ and $\lambda_2 = \lambda_F$: the network may be disconnected by ΔL. We can therefore affirm that the Fiedler's eigenvalue λ_F is related to the minimal norm of the matrix ΔL to disconnect the network.

4.2.1 Steady flow currents

We now investigate the behavior of the perturbed master equation (4.11) at stationary state with input values near to the critical threshold μ_t. Once the iterative scheme finds the critical threshold μ_t we focus on the stationary currents remaining in the network at infinite time and we observe such currents for values around μ_t. Using a manageable example (Fig. 4.3) we compute the exact solution of the perturbed system p^*. We define the stationary current from node i to node j:

$$J_{ij} = \pi_{ij} p_j^* - \pi_{ji} p_i^*$$

We first observe the presence of peaks of currents intensity in correspondence of certain critical input values (Fig. 4.3B); in particular the first peak of currents intensity (Fig. 4.3C) corresponds to the critical threshold found by the iterative scheme μ_t (red-dashed line); then we plot the steady flow currents remaining in the network (Fig. 4.3D) with edge widths proportional to the currents intensity for significant input values: comparing Fig. 4.3D1 and Fig. 4.3D3 we notice a qualitatively different behavior of the stationary currents before and after the critical threshold. The steady flow currents are subject to a dramatic change across the edge value μ_t confirming such value as a reliable turning point for the perturbed master equation dynamics.

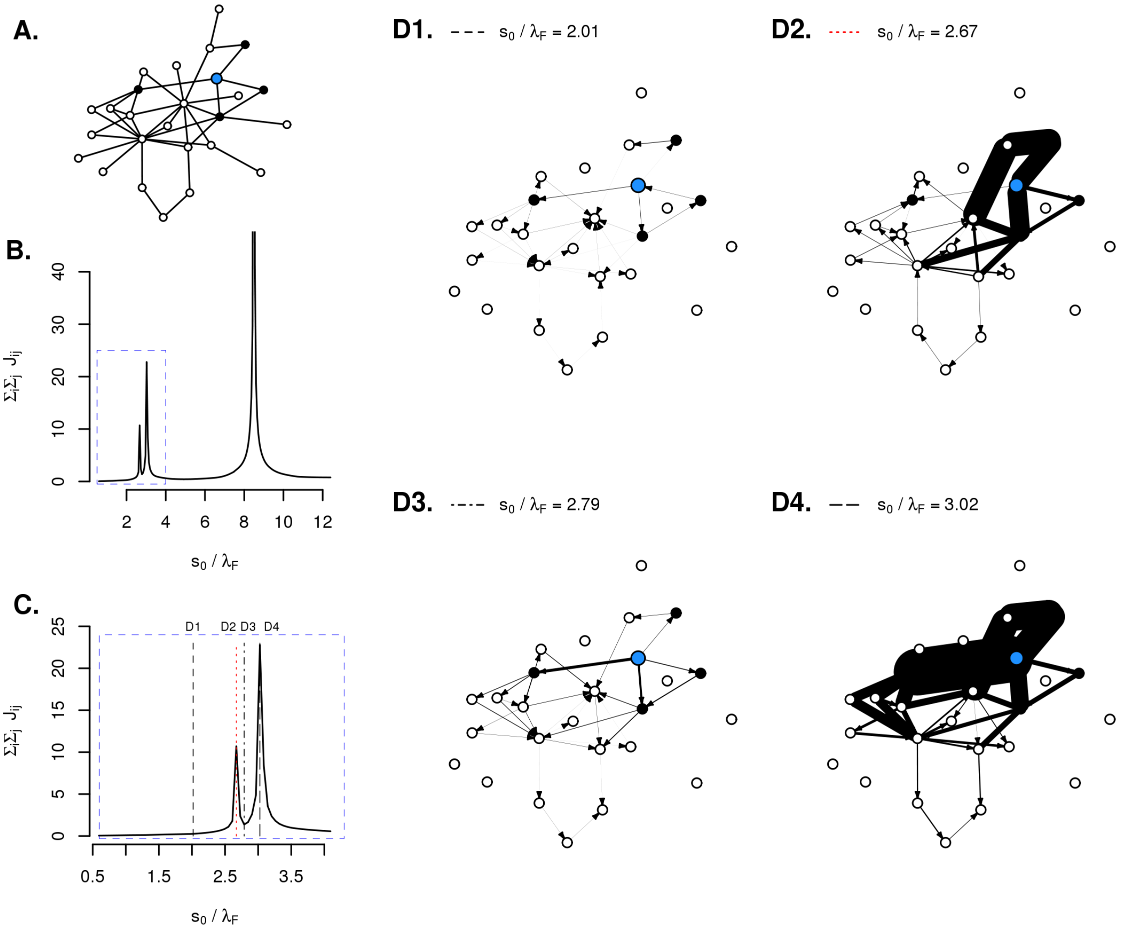


Figure 4.3: **Steady flow currents. A.** Network of 25 nodes extracted from STRING database. 4 nodes are chosen as query nodes and linked to the source/sink node (blue vertex). **B.** Steady flow current intensity ($\sum_i \sum_j J_{ij}$) is plotted vs increasing input intensity s_0 normalized by the Fiedler number of the network λ_F. **C.** Zoom in of the previous. The dashed vertical red line corresponds to the critical threshold. **D.** Steady flow currents remaining in the network at infinite time for increasing values of the input. D2 corresponds to the critical threshold.

As the input value increases above the threshold we notice the presence of other peaks of currents intensity (e.g. Fig. 4.3D4) that represent other turning points for the network dynamics. This interesting behavior is connected to the number of independent loops formed by the stationary currents present on the network. This fact is not a central point in our discussion, however it would require more investigation.

4.2.2 Stability and network measures

Given a a set of synthetic networks (Fig. 4.4) we investigate which perturbations give rise to lower critical thresholds (unstable configurations) and which ones give rise to higher critical thresholds (stable configurations). We remark that the sizes of synthetic networks are in line with the sizes of the biological pathways / subnetworks used for the analysis. We see that for any 2-modules composition (open / closed ring, clique or star) the distribution of critical thresholds in condition A (the two query nodes lie in the same module) is more stable than when condition B holds (the two query nodes lie in the different modules). This observation is related to a sort of "centrality" measure of the perturbation: a perturbation tends to result more stable if confined to only one part of the network (A), while unstablity increases when more network modules are involved in the perturbation (B). In fact we find that the stability or instability of a perturbation is related to the betweenness centrality β_0 of the external node in the extended network with Laplacian matrix $L_0 + \Delta L$ (Fig. 4.4 **e**,**f**,**g**,**h**); β_0 is defined as the fraction of minimal paths connecting any couple of nodes passing from the external node. The higher the perturbation node betweenness the higher the probability to find an unstable configuration. The betweenness of the source-sink node well characterizes the stability of a given configuration on chains (Fig. 4.4 **e**, **f**) while cliques and stars (Fig. 4.4 **g**, **h**) would need more insights from this perspective.

4.2.3 Application to STRING hot subnetwork

On STRING protein-protein interaction network we selected the 25-nodes biggest connected component among the top 100 frequently mutated genes (f is the mutation frequency) in PRAD dataset [81]. The genes involved in such “hot” network for prostate cancer disease are characterized by mutations frequencies that go from a minimum $f = 2\%$ up to a maximum $f = 12\%$ (TP53 and SPOP). As we can see in figure (Fig. 4.5-left) the network has a hub in TP53, that is one of the most studied cancer-related genes [100, 101]. In this case we are interested in understanding which configurations including TP53 cause more unstability to the network information flow. For simplicity we consider only couples of query nodes. It is interesting to notice that the configuration that gives rise to the most unstable network dynamics includes TP53 (the other query node is CDH23), and that at the same time we find stable configurations including TP53 (Fig. 4.5-right).
Even if for a clear biological interpretation would be required more statistical work (e.g. many variables enter the choice of the "hot" subnetwork), from a technical point of view this observation suggests that hubs carrying abnormal content of molecular information can play important roles in the stability regulation of a network information flow depending on the distribution of the other less connected

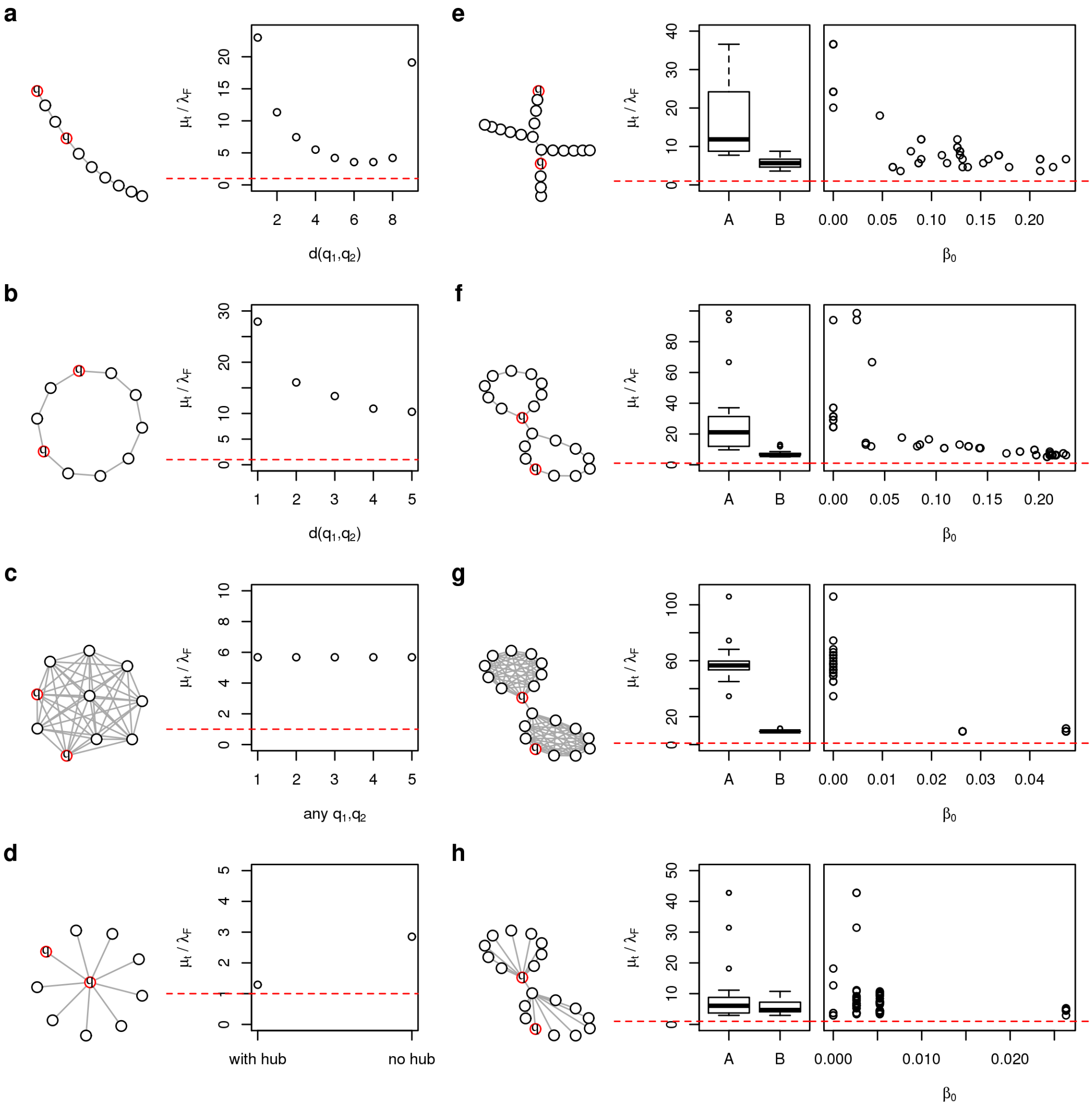


Figure 4.4: **Synthetic Modules.** Stability of synthetic modules perturbed with 2 query nodes measured using the critical threshold (μ_t/λ_F). Network characterization of simple 10-nodes modules (**a**, **b**, **c**, **d**) and their correspondent compositions (**e**, **f**, **g**, **h**). The boxplot labels indicate wether the two query nodes lie in the same module (A) or different ones (B), while the x-axis variable β_0 is the betweenness centrality of the external perturbation node. The horizontal dashed red lines represent the lower bound for the threshold given by the fidelr number λ_F.

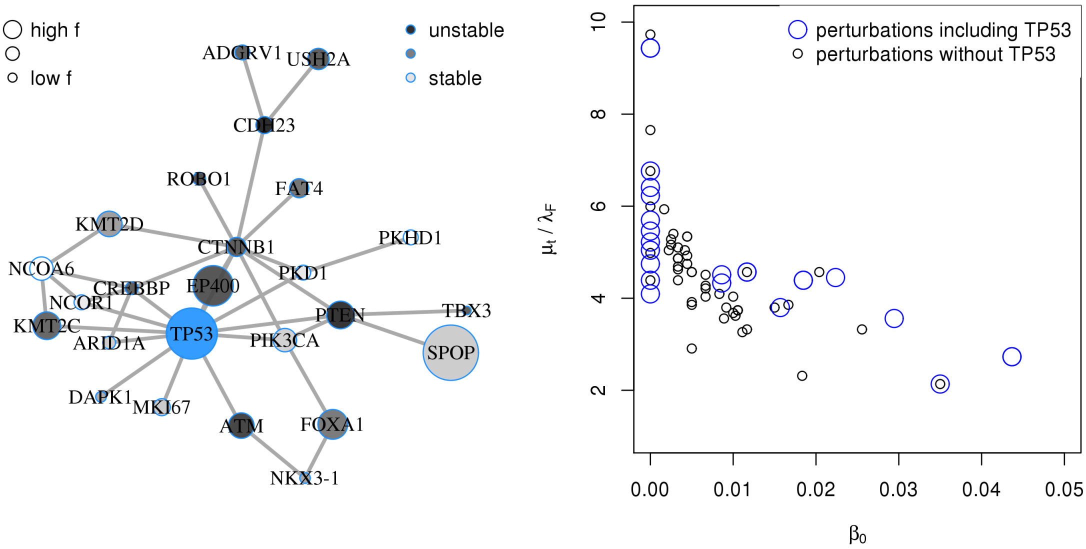


Figure 4.5: **PRAD hot subnetwork.** On STRING P.P.I. Network we selected the 25-nodes biggest connected component among the first 100 frequently mutated genes (f is the mutation frequency) in TCGA dataset. We measured the stability of the all the possible couples of query nodes that include the gene TP53. On the right the plot of the stability (μ_t/λ_F) vs the betweenness centrality β_0 of external node.

query nodes. In other words it is interesting to observe that in this context the stability of a network is related to a collective behavior [102] that is the coordinated local activity of many interdependent components.

Leaving the biological and statistical insights of "hot" subnetwork stability to future work, we now focus on the biological pathways associated to prostate cancer in literature, as they represent more significant targets from a biological point of view.

4.3 Perturbative approach applications to pathway analysis

The definition of a network-based perturbative approach allows to investigate omics data from a novel perspective. In general any kind of omic information can be mapped on a biological network with the purpose of measuring the impact of such information on the network. In the network medicine context [64, 103], a natural application of the proposed method can be summarized in a few steps (Fig. 4.1): first biological networks (such as pathways) must be identified as targets of the application; second the abnormal omic information must be mapped on the target network: the nodes carrying abnormal information (usually deleterious molecular information in the case of a disease) become the query nodes for the extended source/sink model. Such nodes can be mapped with weighted or unweighted edges to the external node: many choices can be taken from this perspective. The last step consists of the application of the iterative scheme in order to find the critical threshold.

In this work, as a proof of principle, we mapped TGCA prostate cancer somatic mutations data on several pathways known in literature to be associated to prostate cancer [100, 101], with the purpose of measuring the impact of such abnormal information on the selected pathways. In this work the pathways are simplified as extracted undirected subgraphs from STRING protein-protein interaction network. The links of the query nodes with the external nodes are undirected, so that the abnormal information is pumped in the network from the external node and can return back through the sinks. We choose to perform both an aggregate analysis and a patient-oriented analysis. For the first one we selected 38 pathways in literature that are known to be associated with prostate cancer and we performed the pathway stability analysis selecting the query nodes from the most frequently mutated genes across the whole dataset. For the second we selected only 4 pathways (out of the 38 previously mentioned) and for the analysis we kept only the patients having enough mutations to perform the stability analysis on at least 1 of the 4 selected pathways. In fact TGCA prostate cancer data is characterized by a very poor number of mutations per patient and so far the molecular characterization of the disease including somatic mutations is still under investigation [100, 101].

4.3.1 PRAD dataset

Prostate adenocarcinoma (PRAD) clinical data were downloaded from the TCGA portal [81]. We focused on the somatic mutation data (collected with the Illumina Genome Analyzer platform) and PRAD RNA sequencing data (GE) (collected with the Illumina HiSeq 2000 RNA Sequencing (Version 2) platform) were downloaded from the TCGA portal for the 261 subjects. Only primary solid tumors (TCGA short letter code "TP") were considered. Both datasets were updated to Entrez Gene [85] identifiers released June 26th 2015.

The somatic mutation dataset presents mutations in 6,898 genes. This dataset was encoded as a binary genes-by-samples matrix where the generic element a_{ij} was set to 1 if the patient j had at least one mutation in gene i, analogously to Hofree[46]. The median number of mutated genes per patient is 30 (out of the 11535 genes of the STRING interactome [80]), but the data are very heterogeneous: we can go from a patient for whom no mutations are registered to a maximum of 734. This fact suggests that a more complete omic analysis would definitively require the integration with more omics layers of information as we show in chapter 3; however we proceed using only somatic mutations in order to see if in at least a considerable percentage of patients we can perform pathway stability analysis.

4.3.2 Aggregate analysis

With the term aggregate analysis we mean that we selected the query nodes from the most frequently mutated genes across the whole dataset (Fig. 4.6). We choose 3 different classes of signal: genes mutated in at least 5 samples (S1), genes mutated in at least 4 samples (S2) and genes mutated in at least 3 samples (S3) (Fig. 4.6A). Then 38 pathways were selected basing ourselves on the literature [100, 101] from an aggregated database including Reactome, Pathway Interaction Database (P.I.D.) and KEGG, with sizes ranging from a minimum of 10 genes to more than 250 genes. In (Fig. 4.6B) we show the number of query nodes per pathway associated with the class of signal. The query nodes corresponding to the signal class were mapped on each pathway (e.g. Fig. 4.6C). It is interesting to notice how even if the selected pathways on average present more signal that non-selected pathways (over 7000 of the human interactome), there are pathways of correspondent sizes that present more query nodes (Fig. 4.6D). We remark that choosing a sample of biologically significant pathways associated to the prostate cancer is not a trivial task; for example one could select the pathways mostly enriched with somatic mutations. However with this particular disease - as previously mentioned - the role of somatic mutations still under investigation [100, 101], so that it would not be clear to which degree the pathways enriched with the most signal are biologically meaningful. We computed the critical threshold for each of the 38 pathways with the 3 different signals (Fig. 4.6E); plotting their distributions we see that on average the thresholds decrease with increasing signal. This fact is expected since increasing the number of query nodes one naturally increases the instability of a fixed network.

The results are shown in figure (Fig. 4.7). We chose to use the weaker signal (S1) involving as query nodes only the genes mutated in 5 or more samples. After computing each critical treshold (μ_t) we divide it by the Fiedler number (λ_F) associated with the correspondent pathway (red vertical lines). This quantity itself is a meaningful measure of pathway instability: the lower the threshold the higher the pathway instability. The critical thresholds could define a natural boundary separating the critical perturbations of the pathways (below the real tresholds) to the less dangerous ones (above the threshold). To better understand the results, for each pathway we randomly resampled the existing signal (S1) and computed the critical threshold for each permutation (horizontal boxplots). An empirical p-value is then defined as the fraction of times in which the permuted thresholds result lower or equal to the real one and the pathways are ordered for increasing p-value. In this way the pathways scoring low p-values are likely to be significantly altered by the somatic mutation data.

These results depend on many different assumptions (selection of pathways, definition of signal), limitations (the analysis is performed only on the STRING interactome, the pathways are considered only as the biggest connected component extractable from STRING PPI, only a small percentage of PPI links is nowadays known) and statistical. The purpose of this analysis is to show that an application of the proposed method is realistic and that the method is capable of capturing the pathway instability returning so far results interesting on the true positive side and not so accurate on the remaining because of the assumptions and limitations just mentioned. However it is interesting to see PI3K-Akt, MAPK and Prostate Cancer pathways as most significant unstable pathways, three of the most studied in the context of prostate cancer disease.

4.3.3 Patient-oriented analysis

We now show the direct application of the iterative scheme to the molecular species × patients TGCA database demonstrating the possibility of patient-oriented analysis. In principle this perspective is very interesting since the application to the patient specific data could avoid unnecessary statistical issues arising in the aggregate analysis and would consider the exact distribution of mutation of each patient as the query nodes. Roughly speaking even if a patient presents mutations that are rare on average, the combined distribution of such rare molecular alterations can lead to high instability on a given pathway. This information is usually considered as noise in an aggregate analysis. On the other hand a limitation of the proposed

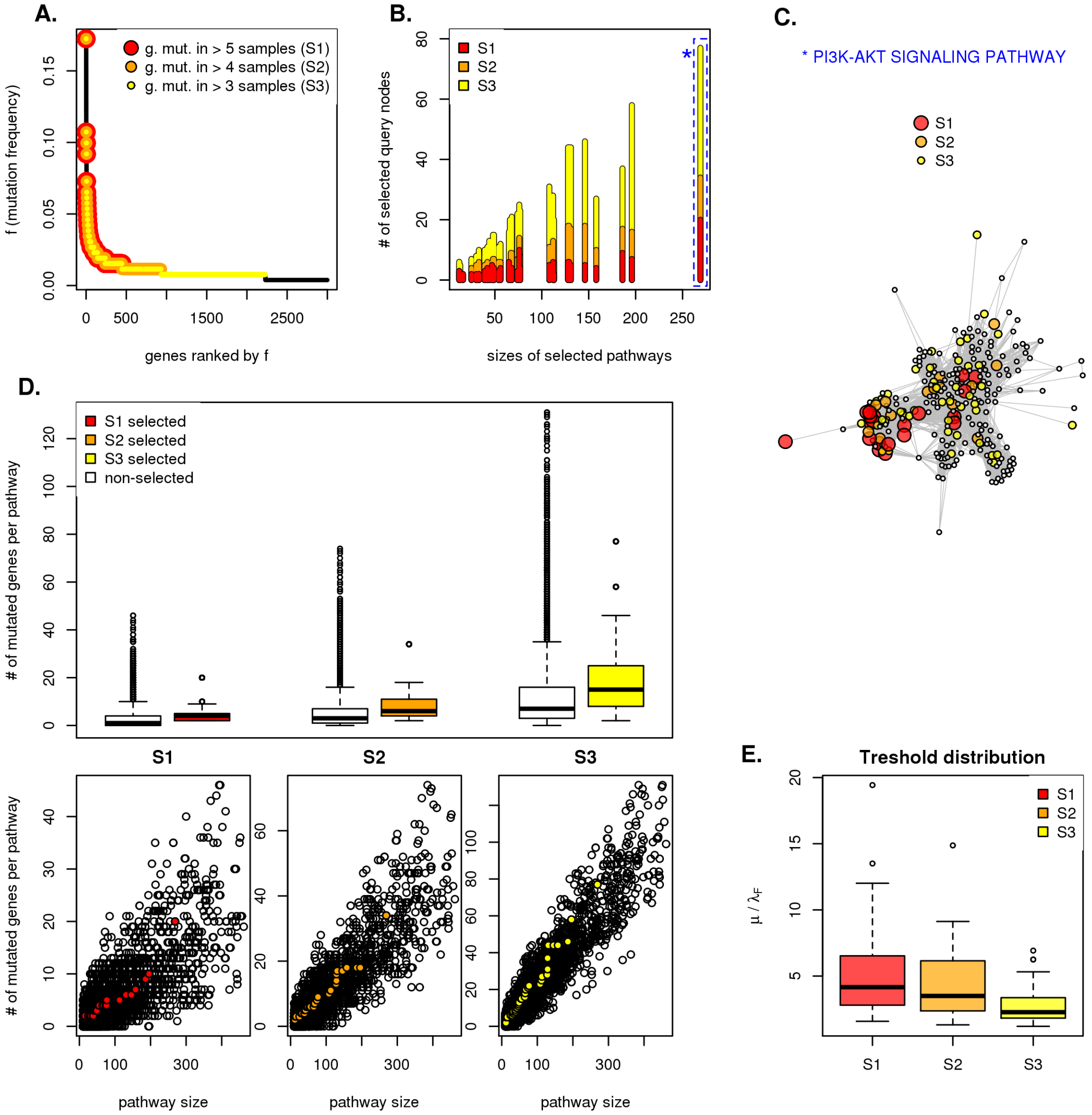


Figure 4.6: **PRAD aggregate signal. A.** Genes mutation frequency plotted by decreasing rank. 3 classes of signal are chosen S1, S2, S3: the query nodes are taken into each of these classes. **B.** Amount of signal over 38 selected pathways known to be invoved in Prostate Cancer disease plotted vs the pathway size. **C.** Example of query nodes on a target pathway correspondent to the 3 different classes of signal. **D.** Signal distribution in the selected pathways compared to non-selected ones. **D.** Critical Treshold distribution over the 3 classes of signal.

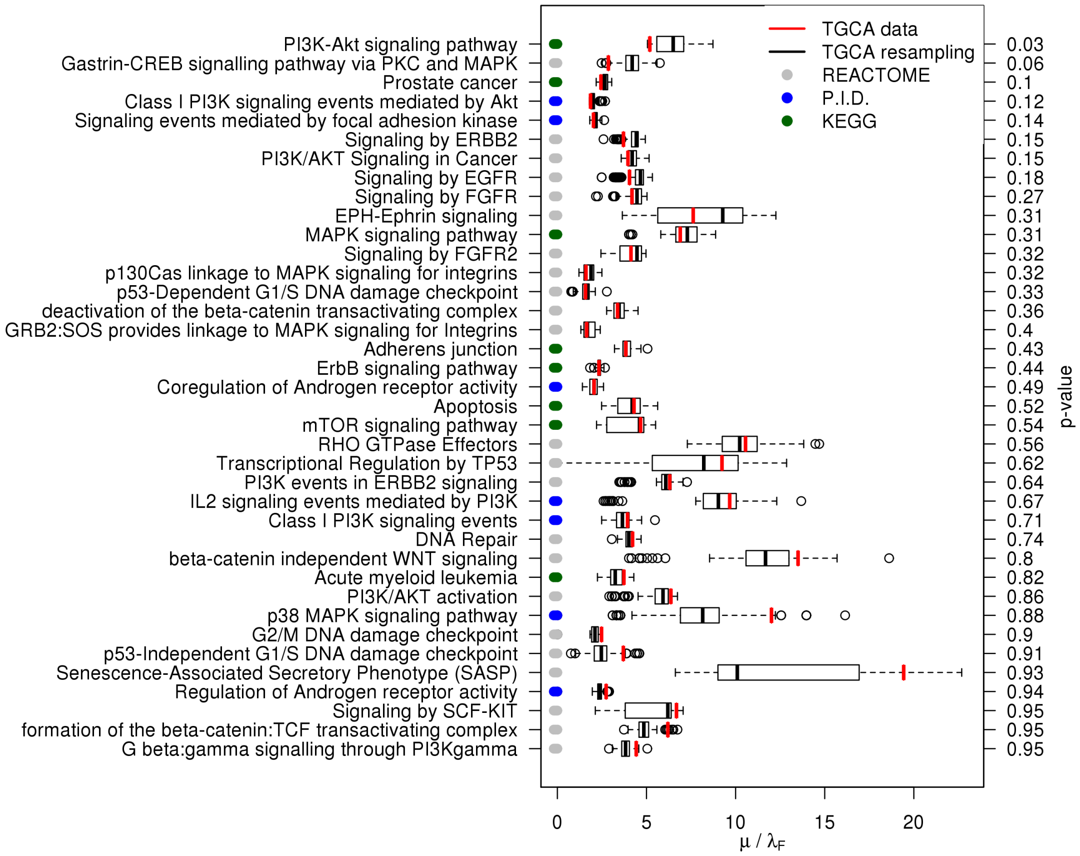


Figure 4.7: **PRAD pathway stability analysis.** The stability of 38 selected pathways is studied. The pathways are ordered from top to bottom for increasing p-values (100 permutations of query nodes).

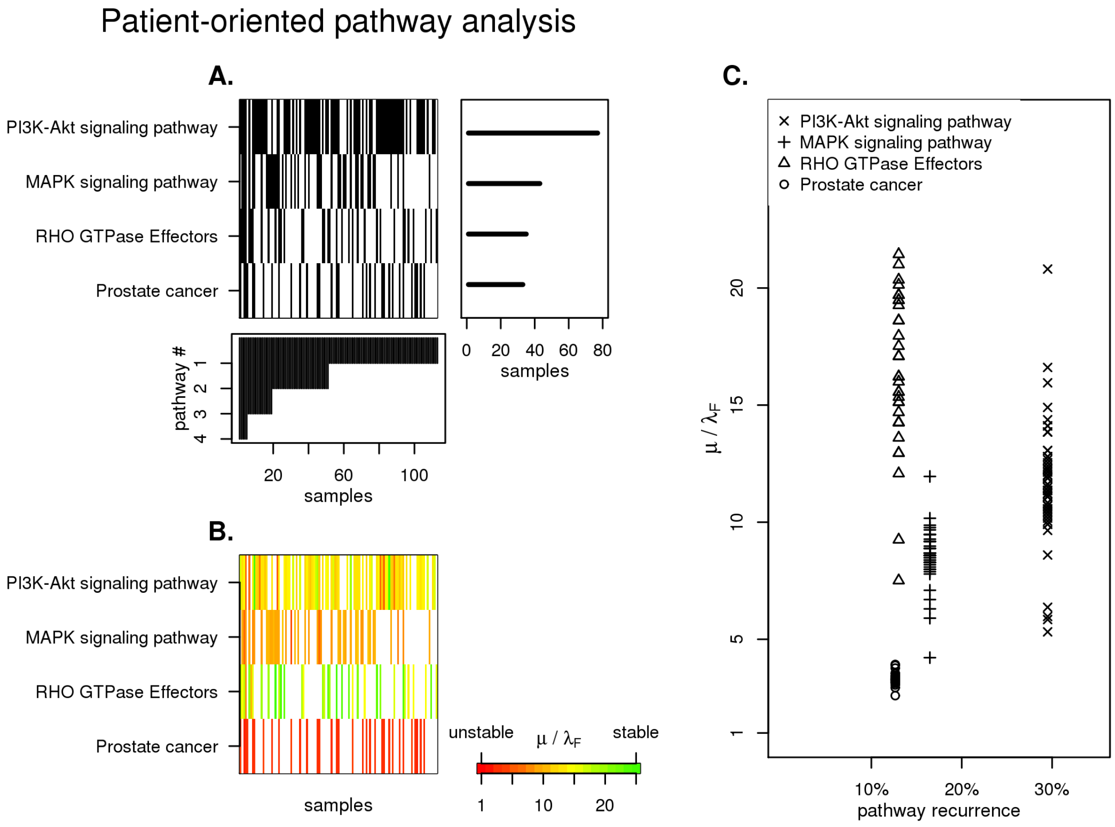


Figure 4.8: **PRAD patient oriented analysis. A.** Patients and pathways available for stability analysis **B.** Treshold distributions per pathway **C.** Treshold distributions per pathway plotted vs pathway recurrence.

approach lies in the unavoidable elimination from the analysis of pathways and patients that present insufficient signal to perform the analysis - given the very low number of mutations per patient.

Always as a proof of principle we considered the 4 pathways for which the perturbative approach was performable for most number of patients (Fig. 4.8A). In this way the dataset was reduced from 261 to 113 patients presenting sufficient signal in at least 1 of the 4 selected pathways. We performed the analysis for each patient measuring the instability of each pathway under the patient-specific mutations.

In this way we can produce pathway-specific or patient-specific profiles giving a picture of molecular pathway instability for the examined disease (Fig. 4.8B). The distribution of critical thresholds vs the recurrence of the pathway (fraction of patients for which the signal is sufficient to perform the analysis) could lead to further analysis (Fig. 4.8C). Prosate cancer pathway results the most unstable pathway however the pathway with the lower recurrence: if a patient has sufficient signal on it the pathway tends to be highly unstable, while for example MAPK and PI3K-Akt present higher recurrence but a more heterogeneous distribution, with a few patients having a much more instability than others.

4.4 Discussion and conclusions

In this chapter we developed a perturbative approach to the master equation with the purpose of measuring how the presence of altered nodes (query nodes) exchanging information with an external node modifies the information flow of the network. As a proof of principle, we applied the stability measurements to both synthetic and PRAD somatic mutations data. In the case of biological networks the query nodes correspond to molecular alterations, but in principle the method can be adapted to measure any kind of network (e.g. social, finantial, transportation, ecological) that is subject to a perturbation.

We defined a theoretical method that measures the impact of a perturbation on a fixed network where the perturbation is a matrix ΔL that defines novel connections between "altered" nodes and the environment. The impact of a perturbation is measured comparing the non-perturbed information flow in the network and the perturbed one that is characterized by the exchange of information with the external node. Such comparison is performed through the definition of an algorithm that recursively shifts from the stationary distribution of the isolated system p_s to the perturbed one $p*$. If the perturbation ΔL is "small" the reccurrence is a contraction and drives p_s into $p*$ in a finite number of steps. On the other hand if ΔL makes the reccurrence have a divergent behavior means that the perturbation macroscopically changed the network information dynamics. In a given configuration of sources / sinks, the input value s_0 corresponding to the shift between convergence and divergence is defined as a critical threshold. We showed that such threshold has the Fiedler number of the network λ_F as a lower bound. An evidence of the intrinsically strong meaning of such critical threshold is emphasized by the presence of peaks of intensity in the steady flow currents of the exact perturbed stationary solution $p*$ in correspondence of the critical input values.

The potential applicability of this method is wide since the method works for any positive Laplacian perturbation matrix ΔL. However some issues require deeper investigation such as the definition of the critical threshold for varying number of query nodes or the introduction of directed / weighted edges. Also a more accurate statistical insight of the relationship between stability of a network and network measures needs to be performed. For example we showed a strong connection between configurations of query nodes having high external node betweenness centrality β_0 and low critical thresholds; however it's clear that even if β_0 may be the driving topological factor associated to the disruptiveness of a perturbation, more insights need to be investigated: the results sensibly changes when the topology of the syinthetic modules varies, and there's evidence that β_0 is not the only network measure associated to instability.

The biological applications show the possibility to apply the perturbative approach to omic datasets both in an aggregate and patient-specific fashion. The results of the analyses show the stability measures of the pathways associated in literature to prostate cancer. In both the aggregate and the patient-oriented analyses we find interesting insights even if the results are still under development.

The future work in this area follows three major guidelines. The first one is the statistical and biological validation of the method, the second one is the computational optimization of the code performing the analysis, and finally the theoretical development of the method in the control theory area. The term "control theory" is frequently used in many disciplines with disparate meanings. Here we mean it in the strict mathematical sense of control theory, an interdisciplinary branch of engineering and mathematics [102, 104]. Control theory asks how to influence the behavior of a dynamical system with appropriately chosen inputs so that the system's output follows a desired trajectory or final state. So in our case we are defining the theoretical tools that would allow us to study in this perspective the effect of changing the query nodes (and therefore the "inputs" of the dynamical system on the network) in order to reach a desired final state $p*$. In terms of biological applications these concepts translate for example to the definition of the sets of molecular alterations that lead to a final state $p*$ that is critical for the network information flow, and therefore likely to be associated with a given disease.

Conclusions and future work

In this work we presented the state of the art of omics and multi-omics data integration methods trying to highlight the most challenging mathematical issues arising from it (Part I). We revised the literature selecting the most promising data integration techniques paying particular attention to multi-omics methods. In fact methods for the analysis of multiple layers of biological information pave the way for a more comprehensive and deeper understanding of biological systems. Indeed, several authors were able to show that the integration of multi-dimensional datasets leads to better results from a statistical and a biological point of view than single layer analyses [14, 19, 23].

In our revision of omics literature we noticed a growing interest around network-based methods. Networks allow to model the intricate cells wiring diagram and to use it as a framework for the integrated analysis of layers of biological information. Such approaches use graphs for modeling and analyzing relationships among omic variables and take advantage of algorithms for graph analysis: in particular, algorithms that propagate information on networks are being proposed in several applications and are often related to actual physical models. In this perspective we noticed that the network propagation algorithm [46, 57] can be derived as a Euler forward implementation of a hydrodynamic model in which an ideal fluid enters the network through the "query nodes", flows along the network edges and exits each node with a constant first order rate.

The theoretical part of the thesis (Part II) was inspired by a work of Mirzaev and Gunawardeena [51] in which after describing all the major results about linear Laplacian dynamics on networks they point out that such dynamics can be formally seen as a chemical master equation described by Van Kampen [56] in which the substance diffusing on the network is the average probability to find the network in a given state at a given time. In this perspective we studied the chemical master equation and constructed a microscopic stochastic model for flow information on the network. We developed the random walk on a network, and under chosen assumptions we discussed the phenomenon accurately. Of course the random walk model is interesting for many reasons - not last its applicability - and it remains fundamentally at the base of the omic tools defined and developed in both the applications (Part III).

Interestingly, the random walk model is a particular case of biochemical network that is discussed exensivly in the work by Elf and Ehremberg [60]. The idea is that, given a biological network, the transitions between states are modeled by linear or non-linear functions of the states. The reason why this generalization could be important relies on the fact that intra or inter omics exchanges could be poorly described by a random walk. The definition of master equations more adequate to omic data modeling and associated modified numerical implementations will be part of our future work.

In both applications of the described models to biological data (Part III) we defined novel network measures that help finding respectively the network modules that are mostly enriched in differential omic information (DEM) and an intrinsic measure of stability associated to a set of altered molecular information on the network. Many considerations and limitations were already treated in the related chapters (3,4).
As a proof of principle, we have applied the proposed methods to prostate ardenocarcinoma (PRAD) data. A deeper investigation of PRAD biology is beyond the scope of our work, nevertheless, we provided interesting biological results. Regarding the DEM method we provide several genes which are very likely to have a role in the different prognostic outcome. In fact, these genes lie in network proximity to genes already associated to PRAD and in regions of the PPI network enriched in mutated and/or differentially expressed genes. In line with the local hypothesis our analysis revealed the existence of a large connected component of genes that are associated with molecular variations (genetic mutations and/or differential expression) between subjects of different prognostic groups. As regards the perturbative approach, the biological applications to the same PRAD dataset show the possibility to apply the perturbative approach to omic datasets both in an aggregate and patient-specific fashion. The results of the analyses show the stability measures of the pathways associated in literature to prostate cancer. In both the aggregate and the patient-oriented analyses we find interesting insights even if the results are still under development. For example in the pathway oriented analysis we register a high instability (low thresholds) associated to the prostate cancer pathways.
In the future work the two methodologies will be further developed and applied to several omic datasets. In particular the DEM discovery is ready to be applied to novel omic datasets both in “horizontal” perspective (e.g. analyze somatic mutations in all available types of cancer) and the “vertical” one (e.g integration of different layers of omic information for patients affected by the same disease). On the other hand the perturbative approach still needs to be statistically validated both in synthetic and real data, then the code performing the analysis needs to be optimized. A particularly interesting future topic is the theoretical development of the perturbative method in the control theory area. Control theory should study the effect of changing the altered nodes in order to reach a desired final state. In terms of biological applications these concepts translate for example to the definition of the sets of molecular alterations that lead to a final state that is critical for the network information flow, and therefore likely to be associated with a given disease.

Published articles

- Bersanelli M, Mosca E, Remondini D, Castellani G, Milanesi L. *"Network Diffusion-based analysis of high-throughput data for the detection of differentially enriched modules."* Scientific reports, vol. 6, p. 34841, 2016.
- Bersanelli M, Mosca E, Remondini D, Giampieri E, Sala C, Castellani G, Milanesi L. *"Methods for the integration of multi-omics data: mathematical aspects."* BMC Bioinformatics, 2016. Suppl 2:15. doi: 10.1186/s12859-015-0857-9. PMID: 26821531. ISSN: 1471-2105. IF 2015: 2.576.
- Sala C, Vitali S, Giampieri E, do Valle IF, Remondini D, Garagnani P, Bersanelli M, Mosca E, Milanesi L, Castellani G. *"Stochastic neutral modeling of the Gut Microbiota's relative species abundance from next generation sequancing data."* BMC Bioinformatics, 2016. 17 Suppl 2:16. doi: 10.1186/s12859-015-0858-8. PMID: 26821617. ISSN: 1471-2105. IF 2015: 2.576.
- Bersanelli M, Dritschel D, Lancellotti C, Poje A. *"Models of interacting pairs of thin, quasi-geostrophic vortices: steady-state solutions and nonlinear stability"*, Geophysical and Astrophysical Fluid Dynamics, DOI: 10.1080/03091929. 2016.1250154.
- Castellani G, Menichetti G, Garagnani P, Giulia Bacalini M, Pirazzini C, Franceschi C, Collino S, Sala C, Remondini D, Giampieri E, Mosca E, Bersanelli M, Vitali S, Valle IF, Liò P, Milanesi L. *"Systems medicine of inflammaging"* Briefings in Bioinformatics, 2015. pii: bbc062 [Epub ahed of print] PMID: 26307062. ISSN: 1477-4054. IF 2015: 7.017.

List of Figures

List of Tables

Bibliography

[1] M. Bersanelli, E. Mosca, D. Remondini, E. Giampieri, C. Sala, G. Castellani, and L. Milanesi, "Methods for the integration of multi-omics data: mathematical aspects.," *BMC bioinformatics*, vol. 17 Suppl 2, p. 15, Jan 2016.

[2] B. Berger, J. Peng, and M. Singh, "Computational solutions for omics data.," *Nat Rev Genet*, vol. 14, pp. 333–346, May 2013.

[3] V. N. Kristensen, O. C. Lingjærde, H. G. Russnes, H. K. M. Vollan, A. Frigessi, and A.-L. Børresen-Dale, "Principles and methods of integrative genomic analyses in cancer.," *Nat Rev Cancer*, vol. 14, pp. 299–313, May 2014.

[4] K.-A. Lê Cao, I. González, and S. Déjean, "integromics: an r package to unravel relationships between two omics datasets," *Bioinformatics*, vol. 25, no. 21, pp. 2855–2856, 2009.

[5] W. Li, S. Zhang, C.-C. Liu, and X. J. Zhou, "Identifying multi-layer gene regulatory modules from multi-dimensional genomic data.," *Bioinformatics*, vol. 28, pp. 2458–2466, Oct 2012.

[6] A.-L. Barabási, N. Gulbahce, and J. Loscalzo, "Network medicine: a network-based approach to human disease.," *Nat Rev Genet*, vol. 12, pp. 56–68, Jan 2011.

[7] J. Skilling, *Data Analysis: A Bayesian Tutorial.* Oxford University Press, 2006.

[8] T. S. Ferguson, "A bayesian analysis of some nonparametric problems," *The annals of statistics*, pp. 209–230, 1973.

[9] D. Heckerman, *A tutorial on learning with Bayesian networks.* Springer, 1998.

[10] O. Gevaert, F. De Smet, D. Timmerman, Y. Moreau, and B. De Moor, "Predicting the prognosis of breast cancer by integrating clinical and microarray data with bayesian networks.," *Bioinformatics*, vol. 22, pp. e184–e190, Jul 2006.

[11] N. Friedman, M. Linial, I. Nachman, and D. Pe'er, "Using bayesian networks to analyze expression data.," *J Comput Biol*, vol. 7, no. 3-4, pp. 601–620, 2000.

[12] U. D. Akavia, O. Litvin, J. Kim, F. Sanchez-Garcia, D. Kotliar, H. C. Causton, P. Pochanard, E. Mozes, L. A. Garraway, and D. Pe'er, "An integrated approach to uncover drivers of cancer.," *Cell*, vol. 143, pp. 1005–1017, Dec 2010.

[13] R. Shen, A. B. Olshen, and M. Ladanyi, “Integrative clustering of multiple genomic data types using a joint latent variable model with application to breast and lung cancer subtype analysis.,” *Bioinformatics*, vol. 25, pp. 2906–2912, Nov 2009.

[14] R. Chari, B. P. Coe, E. A. Vucic, W. W. Lockwood, and W. L. Lam, “An integrative multi-dimensional genetic and epigenetic strategy to identify aberrant genes and pathways in cancer,” *BMC systems biology*, vol. 4, no. 1, p. 67, 2010.

[15] M. R. Aure, I. Steinfeld, L. O. Baumbusch, K. Liestøl, D. Lipson, S. Nyberg, B. Naume, K. K. Sahlberg, V. N. Kristensen, A.-L. Børresen-Dale, O. C. Lingjærde, and Z. Yakhini, “Identifying in-trans process associated genes in breast cancer by integrated analysis of copy number and expression data.,” *PLoS One*, vol. 8, no. 1, p. e53014, 2013.

[16] B.-J. Chen, H. C. Causton, D. Mancenido, N. L. Goddard, E. O. Perlstein, and D. Pe’er, “Harnessing gene expression to identify the genetic basis of drug resistance.,” *Mol Syst Biol*, vol. 5, p. 310, 2009.

[17] E. Mosca and L. Milanesi, “Network-based analysis of omics with multi-objective optimization.,” *Mol Biosyst*, vol. 9, pp. 2971–2980, Dec 2013.

[18] E. Mosca, P. Pelucchi, R. Chen, O. Palumbo, M. Ferrancin, M. Carrella, M. Negrini, B. Neel, R. Reinbold, I. Zucchi, and L. Milanesi, “Network-based integration of protein-silac, mirna and mrna expression data for studying epithelial to mesenchymal transition.,” 2013.

[19] B. Wang, A. M. Mezlini, F. Demir, M. Fiume, Z. Tu, M. Brudno, B. Haibe-Kains, and A. Goldenberg, “Similarity network fusion for aggregating data types on a genomic scale.,” *Nat Methods*, vol. 11, pp. 333–337, Mar 2014.

[20] T. De Bie, L.-C. Tranchevent, L. M. M. van Oeffelen, and Y. Moreau, “Kernel-based data fusion for gene prioritization.,” *Bioinformatics*, vol. 23, pp. i125–i132, Jul 2007.

[21] R. Louhimo and S. Hautaniemi, “Cnamet: an r package for integrating copy number, methylation and expression data,” *Bioinformatics*, vol. 27, no. 6, pp. 887–888, 2011.

[22] C. Meng, B. Kuster, A. C. Culhane, and A. M. Gholami, “A multivariate approach to the integration of multi-omics datasets.,” *BMC Bioinformatics*, vol. 15, p. 162, 2014.

[23] Y. Liu, V. Devescovi, S. Chen, and C. Nardini, “Multilevel omic data integration in cancer cell lines: advanced annotation and emergent properties.,” *BMC Syst Biol*, vol. 7, p. 14, 2013.

[24] J. Trygg and S. Wold, “Orthogonal projections to latent structures (o-pls),” *Journal of chemometrics*, vol. 16, no. 3, pp. 119–128, 2002.

[25] R. Rosipal and L. J. Trejo, "Kernel partial least squares regression in reproducing kernel hilbert space," *The Journal of Machine Learning Research*, vol. 2, pp. 97–123, 2002.

[26] M. Bylesjö, D. Eriksson, M. Kusano, T. Moritz, and J. Trygg, "Data integration in plant biology: the o2pls method for combined modeling of transcript and metabolite data.," *Plant J*, vol. 52, pp. 1181–1191, Dec 2007.

[27] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning*. Springer, 2013.

[28] C. E. Antoniak, "Mixtures of dirichlet processes with applications to bayesian nonparametric problems," *The annals of statistics*, pp. 1152–1174, 1974.

[29] R. S. Savage, Z. Ghahramani, J. E. Griffin, B. J. de la Cruz, and D. L. Wild, "Discovering transcriptional modules by bayesian data integration.," *Bioinformatics*, vol. 26, pp. i158–i167, Jun 2010.

[30] P. Kirk, J. E. Griffin, R. S. Savage, Z. Ghahramani, and D. L. Wild, "Bayesian correlated clustering to integrate multiple datasets.," *Bioinformatics*, vol. 28, pp. 3290–3297, Dec 2012.

[31] Y. Yuan, R. S. Savage, and F. Markowetz, "Patient-specific data fusion defines prognostic cancer subtypes.," *PLoS Comput Biol*, vol. 7, p. e1002227, Oct 2011.

[32] C. Huttenhower, K. T. Mutungu, N. Indik, W. Yang, M. Schroeder, J. J. Forman, O. G. Troyanskaya, and H. A. Coller, "Detailing regulatory networks through large scale data integration.," *Bioinformatics*, vol. 25, pp. 3267–3274, Dec 2009.

[33] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[34] P. J. Green and S. Richardson, "Modelling heterogeneity with and without the dirichlet process," *Scandinavian journal of statistics*, vol. 28, no. 2, pp. 355–375, 2001.

[35] N. Tuncbag, S. McCallum, S.-S. C. Huang, and E. Fraenkel, "Steinernet: a web server for integrating 'omic' data to discover hidden components of response pathways.," *Nucleic Acids Res*, vol. 40, pp. W505–W509, Jul 2012.

[36] Y. Cun and H. Fröhlich, "netclass: an r-package for network based, integrative biomarker signature discovery.," *Bioinformatics*, vol. 30, pp. 1325–1326, May 2014.

[37] I. Merelli, P. Lió, and L. Milanesi, "Nuchart: an r package to study gene spatial neighbourhoods with multi-omics annotations.," *PLoS One*, vol. 8, no. 9, p. e75146, 2013.

[38] N. L. van Berkum, E. Lieberman-Aiden, L. Williams, M. Imakaev, A. Gnirke, L. A. Mirny, J. Dekker, and E. S. Lander, “Hi-c: a method to study the three-dimensional architecture of genomes.,” *J Vis Exp*, no. 39, 2010.

[39] S. Aerts, D. Lambrechts, S. Maity, P. Van Loo, B. Coessens, F. De Smet, L.-C. Tranchevent, B. De Moor, P. Marynen, B. Hassan, P. Carmeliet, and Y. Moreau, “Gene prioritization through genomic data fusion.,” *Nat Biotechnol*, vol. 24, pp. 537–544, May 2006.

[40] G. Menichetti, D. Remondini, P. Panzarasa, R. J. Mondragón, and G. Bianconi, “Weighted multiplex networks.,” *PLoS One*, vol. 9, no. 6, p. e97857, 2014.

[41] G. Castellani, N. Intrator, and D. Remondini, “Systems biology and brain activity in neuronal pathways by smart device and advanced signal processing,” *Frontiers in genetics*, vol. 5, 2014.

[42] G. Menichetti, D. Remondini, and G. Bianconi, “Correlations between weights and overlap in ensembles of weighted multiplex networks.,” *Phys Rev E Stat Nonlin Soft Matter Phys*, vol. 90, p. 062817, Dec 2014.

[43] X. Dong, P. Frossard, P. Vandergheynst, and N. Nefedov, “Clustering with multi-layer graphs: A spectral perspective,” *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, vol. 60, no. 11, pp. 5820–5831, 2012.

[44] X. Wang, N. Gulbahce, and H. Yu, “Network-based methods for human disease gene prediction.,” *Brief Funct Genomics*, vol. 10, pp. 280–293, Sep 2011.

[45] R. I. Kondor and J. Lafferty, “Diffusion kernels on graphs and other discrete input spaces,” in *ICML*, vol. 2, pp. 315–322, 2002.

[46] M. Hofree, J. P. Shen, H. Carter, A. Gross, and T. Ideker, “Network-based stratification of tumor mutations.,” *Nat Methods*, vol. 10, pp. 1108–1115, Nov 2013.

[47] F. Vandin, E. Upfal, and B. J. Raphael, “Algorithms for detecting significantly mutated pathways in cancer.,” *J Comput Biol*, vol. 18, pp. 507–522, Mar 2011.

[48] Y. Qi, Y. Suhail, Y.-y. Lin, J. D. Boeke, and J. S. Bader, “Finding friends and enemies in an enemies-only network: a graph diffusion kernel for predicting novel genetic interactions and co-complex membership from yeast genetic interactions,” *Genome research*, vol. 18, no. 12, pp. 1991–2004, 2008.

[49] P. G. Doyle and J. L. Snell, “Random walks and electric networks,” *AMC*, vol. 10, p. 12, 1984.

[50] S. Suthram, A. Beyer, R. M. Karp, Y. Eldar, and T. Ideker, “eqed: an efficient method for interpreting eqtl associations using protein networks.,” *Mol Syst Biol*, vol. 4, p. 162, 2008.

[51] I. Mirzaev and J. Gunawardena, “Laplacian dynamics on general graphs.,” *Bull Math Biol*, vol. 75, pp. 2118–2149, Nov 2013.

[52] Y. Li and J. C. Patra, "Genome-wide inferring gene-phenotype relationship by walking on the heterogeneous network.," *Bioinformatics*, vol. 26, pp. 1219–1224, May 2010.

[53] C. J. Vaske, S. C. Benz, J. Z. Sanborn, D. Earl, C. Szeto, J. Zhu, D. Haussler, and J. M. Stuart, "Inference of patient-specific pathway activities from multi-dimensional cancer genomics data using paradigm.," *Bioinformatics*, vol. 26, pp. i237–i245, Jun 2010.

[54] R Development Core Team, *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2008. ISBN 3-900051-07-0.

[55] MATLAB, *version 7.10.0 (R2010a).* Natick, Massachusetts: The MathWorks Inc., 2010.

[56] N. G. van Kampen, *Stochastic Processes in Physics and Chemistry.* 1992.

[57] O. Vanunu, O. Magger, E. Ruppin, T. Shlomi, and R. Sharan, "Associating genes and protein complexes with disease via network propagation.," *PLoS Comput Biol*, vol. 6, p. e1000641, Jan 2010.

[58] I. Volkov, J. R. Banavar, S. P. Hubbell, and A. Maritan, "Patterns of relative species abundance in rainforests and coral reefs," *Nature*, vol. 450, no. 7166, pp. 45–49, 2007.

[59] C. Sala, S. Vitali, E. Giampieri, Ì. F. do Valle, D. Remondini, P. Garagnani, M. Bersanelli, E. Mosca, L. Milanesi, and G. Castellani, "Stochastic neutral modelling of the gut microbiota's relative species abundance from next generation sequencing data," *BMC bioinformatics*, vol. 17, no. Suppl 2, p. 16, 2016.

[60] J. Elf and M. Ehrenberg, "Fast evaluation of fluctuations in biochemical networks with the linear noise approximation.," *Genome research*, vol. 13, pp. 2475–2484, Nov 2003.

[61] M. Bersanelli, E. Mosca, D. Remondini, G. Castellani, and L. Milanesi, "Network diffusion-based analysis of high-throughput data for the detection of differentially enriched modules.," *Scientific reports*, vol. 6, p. 34841, Oct 2016.

[62] L. H. Hartwell, J. J. Hopfield, S. Leibler, and A. W. Murray, "From molecular to modular cell biology.," *Nature*, vol. 402, pp. C47–C52, Dec 1999.

[63] R. Sharan, I. Ulitsky, and R. Shamir, "Network-based prediction of protein function," *Molecular systems biology*, vol. 3, no. 1, p. 88, 2007.

[64] G. C. Castellani, G. Menichetti, P. Garagnani, M. Giulia Bacalini, C. Pirazzini, C. Franceschi, S. Collino, C. Sala, D. Remondini, E. Giampieri, E. Mosca, M. Bersanelli, S. Vitali, I. F. d. Valle, P. Liò, and L. Milanesi, "Systems medicine of inflammaging.," *Brief Bioinform*, Aug 2015.

[65] S. D. Ghiassian, J. Menche, and A.-L. Barabási, “A disease module detection (diamond) algorithm derived from a systematic analysis of connectivity patterns of disease proteins in the human interactome.,” *PLoS Comput Biol*, vol. 11, p. e1004120, Apr 2015.

[66] M. D. M. Leiserson, F. Vandin, H.-T. Wu, J. R. Dobson, J. V. Eldridge, J. L. Thomas, A. Papoutsaki, Y. Kim, B. Niu, M. McLellan, M. S. Lawrence, A. Gonzalez-Perez, D. Tamborero, Y. Cheng, G. A. Ryslik, N. Lopez-Bigas, G. Getz, L. Ding, and B. J. Raphael, “Pan-cancer network analysis identifies combinations of rare somatic mutations across pathways and protein complexes.,” *Nat Genet*, vol. 47, pp. 106–114, Feb 2015.

[67] E. O. Paull, D. E. Carlin, M. Niepel, P. K. Sorger, D. Haussler, and J. M. Stuart, “Discovering causal pathways linking genomic events to transcriptional states using tied diffusion through interacting events (tiedie).,” *Bioinformatics*, vol. 29, pp. 2757–2764, Nov 2013.

[68] A. Lan, I. Y. Smoly, G. Rapaport, S. Lindquist, E. Fraenkel, and E. Yeger-Lotem, “Responsenet: revealing signaling and regulatory networks linking genetic and transcriptomic screening data.,” *Nucleic Acids Res*, vol. 39, pp. W424–W429, Jul 2011.

[69] Y.-Q. Qiu, S. Zhang, X.-S. Zhang, and L. Chen, “Detecting disease associated modules and prioritizing active genes based on high throughput data.,” *BMC Bioinformatics*, vol. 11, p. 26, 2010.

[70] K. Mitra, A.-R. Carvunis, S. K. Ramesh, and T. Ideker, “Integrative approaches for finding modular structure in biological networks,” *Nature Reviews Genetics*, vol. 14, no. 10, pp. 719–732, 2013.

[71] M. Kotlyar, C. Pastrello, F. Pivetta, A. L. Sardo, C. Cumbaa, H. Li, T. Naranian, Y. Niu, Z. Ding, F. Vafaee, *et al.*, “In silico prediction of physical protein interactions and characterization of interactome orphans,” *Nature methods*, vol. 12, no. 1, pp. 79–84, 2015.

[72] G. Lauc, A. Essafi, J. E. Huffman, C. Hayward, A. Knezevic, J. J. Kattla, O. Polasek, O. Gornik, V. Vitart, J. L. Abrahams, *et al.*, “Genomics meets glycomics-the first gwas study of human n-glycome identifies hnf1alpha as a master regulator of plasma protein fucosylation,” *PLoS Genet*, vol. 6, no. 12, p. e1001256, 2010.

[73] M. E. Stokes, M. M. Barmada, M. I. Kamboh, and S. Visweswaran, “The application of network label propagation to rank biomarkers in genome-wide alzheimer’s data.,” *BMC Genomics*, vol. 15, p. 282, 2014.

[74] Y. Qian, S. Besenbacher, T. Mailund, and M. H. Schierup, “Identifying disease associated genes by network propagation,” *BMC systems biology*, vol. 8, no. Suppl 1, p. S6, 2014.

[75] N. Gulbahce, H. Yan, A. Dricot, M. Padi, D. Byrdsong, R. Franchi, D.-S. Lee, O. Rozenblatt-Rosen, J. C. Mar, M. A. Calderwood, *et al.*, “Viral perturbations of host networks reflect disease etiology,” *PLoS Comput Biol*, vol. 8, no. 6, p. 1002531, 2012.

[76] E. Mosca, R. Alfieri, and L. Milanesi, “Diffusion of information throughout the host interactome reveals gene expression variations in network proximity to target proteins of hepatitis c virus,” *PloS one*, vol. 9, no. 12, p. e113660, 2014.

[77] Y. Cun and H. Fröhlich, “Network and data integration for biomarker signature discovery via network smoothed t-statistics.,” *PLoS One*, vol. 8, no. 9, p. e73074, 2013.

[78] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, “Learning with local and global consistency,” *Advances in neural information processing systems*, vol. 16, no. 16, pp. 321–328, 2004.

[79] B. Vogelstein, N. Papadopoulos, V. E. Velculescu, S. Zhou, L. A. Diaz, Jr, and K. W. Kinzler, “Cancer genome landscapes.,” *Science*, vol. 339, pp. 1546–1558, Mar 2013.

[80] D. Szklarczyk, A. Franceschini, S. Wyder, K. Forslund, D. Heller, J. Huerta-Cepas, M. Simonovic, A. Roth, A. Santos, K. P. Tsafou, M. Kuhn, P. Bork, L. J. Jensen, and C. von Mering, “String v10: protein-protein interaction networks, integrated over the tree of life.,” *Nucleic Acids Res*, vol. 43, pp. D447–D452, Jan 2015.

[81] “The cancer genome atlas data portal.”

[82] A. Subramanian, P. Tamayo, V. K. Mootha, S. Mukherjee, B. L. Ebert, M. A. Gillette, A. Paulovich, S. L. Pomeroy, T. R. Golub, E. S. Lander, and J. P. Mesirov, “Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles.,” *Proc Natl Acad Sci U S A*, vol. 102, pp. 15545–15550, Oct 2005.

[83] D. Hanahan and R. A. Weinberg, “Hallmarks of cancer: the next generation.,” *Cell*, vol. 144, pp. 646–674, Mar 2011.

[84] V. G. Tusher, R. Tibshirani, and G. Chu, “Significance analysis of microarrays applied to the ionizing radiation response.,” *Proc Natl Acad Sci U S A*, vol. 98, pp. 5116–5121, Apr 2001.

[85] G. R. Brown, V. Hem, K. S. Katz, M. Ovetsky, C. Wallin, O. Ermolaeva, I. Tolstoy, T. Tatusova, K. D. Pruitt, D. R. Maglott, and T. D. Murphy, “Gene: a gene-centered information resource at ncbi.,” *Nucleic Acids Res*, vol. 43, pp. D36–D42, Jan 2015.

[86] T. Rolland, M. Tasan, B. Charloteaux, S. J. Pevzner, Q. Zhong, N. Sahni, S. Yi, I. Lemmens, C. Fontanillo, R. Mosca, A. Kamburov, S. D. Ghiassian, X. Yang, L. Ghamsari, D. Balcha, B. E. Begg, P. Braun, M. Brehme, M. P. Broly, A.-R. Carvunis, D. Convery-Zupan, R. Corominas, J. Coulombe-Huntington, E. Dann, M. Dreze, A. Dricot, C. Fan, E. Franzosa, F. Gebreab, B. J. Gutierrez, M. F. Hardy, M. Jin, S. Kang, R. Kiros, G. N. Lin, K. Luck, A. MacWilliams, J. Menche, R. R. Murray, A. Palagi, M. M. Poulin, X. Rambout, J. Rasla, P. Reichert, V. Romero, E. Ruyssinck, J. M. Sahalie, A. Scholz, A. A. Shah, A. Sharma, Y. Shen, K. Spirohn, S. Tam, A. O. Tejeda, S. A.

Trigg, J.-C. Twizere, K. Vega, J. Walsh, M. E. Cusick, Y. Xia, A.-L. Barabási, L. M. Iakoucheva, P. Aloy, J. De Las Rivas, J. Tavernier, M. A. Calderwood, D. E. Hill, T. Hao, F. P. Roth, and M. Vidal, “A proteome-scale map of the human interactome network.,” *Cell*, vol. 159, pp. 1212–1226, Nov 2014.

[87] P. M. Pierorazio, P. C. Walsh, A. W. Partin, and J. I. Epstein, “Prognostic gleason grade grouping: data based on the modified gleason scoring system.,” *BJU Int*, vol. 111, pp. 753–760, May 2013.

[88] P. Langfelder and S. Horvath, “Wgcna: an r package for weighted correlation network analysis,” *BMC bioinformatics*, vol. 9, no. 1, p. 559, 2008.

[89] M. D. Robinson and A. Oshlack, “A scaling normalization method for differential expression analysis of rna-seq data.,” *Genome Biol*, vol. 11, no. 3, p. R25, 2010.

[90] M. D. Robinson, D. J. McCarthy, and G. K. Smyth, “edger: a bioconductor package for differential expression analysis of digital gene expression data.,” *Bioinformatics*, vol. 26, pp. 139–140, Jan 2010.

[91] M. E. Ritchie, B. Phipson, D. Wu, Y. Hu, C. W. Law, W. Shi, and G. K. Smyth, “limma powers differential expression analyses for rna-sequencing and microarray studies.,” *Nucleic Acids Res*, vol. 43, p. e47, Apr 2015.

[92] Y. Xiao, T.-H. Hsiao, U. Suresh, H.-I. H. Chen, X. Wu, S. E. Wolf, and Y. Chen, “A novel significance score for gene selection and ranking.,” *Bioinformatics*, vol. 30, pp. 801–807, Mar 2014.

[93] L. Y. Geer, A. Marchler-Bauer, R. C. Geer, L. Han, J. He, S. He, C. Liu, W. Shi, and S. H. Bryant, “The ncbi biosystems database.,” *Nucleic Acids Res*, vol. 38, pp. D492–D496, Jan 2010.

[94] X. Wang, C. Terfve, J. C. Rose, and F. Markowetz, “Htsanalyzer: an r/bioconductor package for integrated network analysis of high-throughput screens.,” *Bioinformatics*, vol. 27, pp. 879–880, Mar 2011.

[95] X. Wang, C. Terfve, J. C. Rose, and F. Markowetz, “Htsanalyzer: an r/bioconductor package for integrated network analysis of high-throughput screens.,” *Bioinformatics*, vol. 27, pp. 879–880, Mar 2011.

[96] “Biodigitalvalley: Proteinquest, a web based platform for the mining of medline papers..”

[97] R. Tibshirani, G. Chu, B. Narasimhan, and J. Li, *samr: SAM: Significance Analysis of Microarrays*, 2011. R package version 2.0.

[98] P. Khatri, M. Sirota, and A. J. Butte, “Ten years of pathway analysis: current approaches and outstanding challenges.,” *PLoS Comput Biol*, vol. 8, no. 2, p. e1002375, 2012.

[99] Z. Szallasi, J. Stelling, and V. Periwal, “System modeling in cellular biology,” *From Concepts to*, 2006.

[100] C. G. A. R. Network, "The molecular taxonomy of primary prostate cancer.," *Cell*, vol. 163, pp. 1011–1025, Nov 2015.

[101] E. Mazaris and A. Tsiotras, "Molecular pathways in prostate cancer.," *Nephro-urology monthly*, vol. 5, pp. 792–800, Jul 2013.

[102] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Reviews of modern physics*, vol. 74, no. 1, p. 47, 2002.

[103] J. C. Nacher, B. Keith, and J.-M. Schwartz, "Network medicine analysis of chondrocyte proteins towards new treatments of osteoarthritis.," *Proceedings. Biological sciences*, vol. 281, p. 20132907, Mar 2014.

[104] G. Yan, G. Tsekenis, B. Barzel, J.-J. Slotine, Y.-Y. Liu, and A.-L. Barabási, "Spectrum of controlling and observing complex networks," *Nature Physics*, vol. 11, no. 9, pp. 779–786, 2015.