Regulus, a transcriptional regulatory networks inference tool based on Semantic Web technologies

Marine Louarn[1,2,✉], Guillaume Collet[1], Eve Barre[1], Thierry Fest[2,3], Olivier Dameron[1], Anne Siegel[1], and Fabrice Chatonnet[2,3,✉]

[1]Univ Rennes, CNRS, Inria, IRISA - UMR 6074, F-35000 Rennes, France
[2]UMR_S 1236, Université Rennes 1, INSERM, Etablissement Français du Sang, F-35000 Rennes, France
[3]Laboratoire d'Hématologie, Pôle de Biologie, CHU de Rennes, F-35033 Rennes, France.

Motivation: Transcriptional regulation -a major field of investigation in life science- is performed by binding of specialized proteins called transcription factors (TF) to DNA in specific, context-dependent regulatory regions, leading to either activation or inhibition of gene expression. Relations between TF, regions and genes can be described as regulatory networks, which are basically knowledge graphs containing the relationships between the different entities. Current methods of transcriptional regulatory networks inference rarely use information about TF binding or regulatory regions, often require a large number of samples and most of time do not indicate if the TF-gene relation is an activation or an inhibition. The resulting networks may then contain inconsistent relations and the methods are not applicable for common experimental or clinical settings, where the number of samples is limited. Therefore, based on our previous experience of formalizing the Regulatory Circuits data-sets with Semantic Web Technologies, we decided to create a new tool for transcriptional networks inference, that could solve these issues.
Results: Our tool, *Regulus*, provides candidate signed TF-gene relations computed from gene expressions, regulatory region activities and TF binding sites data, together with the genomic location of all entities. After creating expressions and activities patterns, data are integrated into a RDF endpoint. A dedicated SPARQL query retrieves all potential TF-region relations for a given gene expression pattern. These ternary TF-region-gene pattern relations are then filtered and signed using a logical consistency check translated from biological knowledge. *Regulus* compares favorably to its closest network inference method, provides signs which are consistent with public databases and, when applied to real biological data, identifies both known and potential new regulators. We also provide several means to more stringently filter the output regulators. Altogether, we propose a new tool devoted to transcriptional network inference in settings where samples are scarce and cell populations may be closely related.
Availability: The *Regulus* package is available at `https://gitlab.com/gcollet/regulus`

Activity patterns | Logical consistency | Few samples | Related cell types
Correspondence: *`fabrice.chatonnet@chu-rennes.fr`*

1. Introduction

The biology of gene regulation Gene expression regulation (also called *transcriptional regulation*) is a major field of investigation in life science. It allows a better understanding of major processes such as cell differentiation (how one or several effective cell types are generated from a common progenitor cell), cell identity (how gene expression is used to define and maintain a specific cell type) and cell transformation (how altered gene expression can lead to cell death or cancer) (1).

In gene regulatory networks, regulators are specialized proteins called transcription factors (TF). TF bind DNA at a definite sequence (called a binding motif or binding site) in specific regulatory regions. TF binding will then initiate a cascade of molecular events eventually leading to the regulation (induction or inhibition) of the target gene expression. Regulatory regions are located in non-coding DNA from 0 to several mega-bases from their target genes (2), and have to be in an accessible 3D conformation to allow the regulation (3). Therefore, chromatin accessibility has a major role on the regulation as it constrains transcription factor binding to DNA: a regulatory region in a "closed" conformation will inhibit any potential regulation for a TF with a binding site inside it.

As evidenced in several recent publications (4–6), gene regulation is extremely context-dependent. For example, pathological processes such as cancer can disturb the transcriptional regulatory networks by modifying regulatory regions accessibility or location, by modifying TF expression or by introducing non-coding mutations - thereby modifying TF fixation abilities (7). Information about regulatory regions and TF expression is therefore essential to build reliable context-dependent gene regulatory networks and predict the effect of genome perturbations observed in pathological contexts, paving the way to personalized treatments.

One key limitation in clinical settings is the reduced number of samples (usually fewer than 20) which can be analyzed to create a patient-specific transcriptional network. Moreover, the compared cell types are typically very closely related, as cells evolve by steps and retain most properties of their cell of origin, which makes the observation of their differences difficult. Studying the dynamics of gene expression changes between each cell types - rather than the static properties of each step - allows for a better description of cellular transitions, which may be promoted or blocked for therapeutic purposes. It relies only on the differences between cell types and considers that TF-gene relations are constant in all sam-

Method name	Reference	Data				Data Normalization			Graph		Type of implementation	Other
		Genes	Regions	TF	Other	2 level discretization	multi-level discretization	continuous	Scored	Signed		
REVEAL	(12)	a(t)				x					Algo	
RelNet	(13)	a(t)				x			x		Algo	
BANJO	(14)	a(t)					x				Algo	
NIR	(15)	a(t)					x				Algo	
ARCANE	(16)	a(t)				x			x		Algo	
TSNI	(17)	a(t)					x		x		Algo	Focus on 1 gene
COALESCE	(18)	a(t)		BS*	nucleosome positioning*, evolutionary conservation*		x		x		C++ implementation & web interface	
DISTILLER	(19)	a		BS				x	x		integration: itself mining	Co-expressed genes
Mix-CLR	(20)	a(t)					x		x		Algo	
TIGRESS	(21)	a(t)					x		x		Matlab implementation	
iRafNet	(22)	a(t)*, a* Knok-down*		BS*	interaction protein-protein*		x		x		R implementation	
Regulatory Circuits	(10)		a	BS			x		x		Workflow	
SINCERITIES	(23)	a(t)						x	x	x	Algo	
PoLoBag	(24)	a(t)						x	x	x	Algo	

Table 1: Review of different network inference methods. a = activity, a(t) time series of the activity, * optional, BS = TF binding site, Algo = description of the algorithm without implementation

ples: a same TF regulates the same gene with the same effect (activation or inhibition). Thus by considering the dynamic transitions and by applying some consistency checking between cell types, one might be able to infer TF functions (activation or inhibition) in each relation. This is instrumental in a therapeutic perspective, in order to design targeted drugs which may potentiate or inhibit specific regulators. From a methodological perspective, this requires methods allowing to build dynamic gene regulation networks by taking into account regulatory regions and their accessibility, together with TF expression and binding and able to output "signed" TF-gene relations.

Use case: regulation driving B cells differentiation As an illustration, let us consider the biological case study of differentiation of B cells into antibody producing cells. This process involves several transitions between closely related cell types, which are finely regulated by genetic networks (8). Their deregulation can lead to immunodeficiencies, autoimmune diseases and hematological malignancies, among which follicular lymphoma which has high prevalence and is considered incurable, due to a high rate of relapse, resistance to treatments and a high inter-patient heterogeneity.

Few large scale studies have been performed to understand B cell normal and pathological transition steps of differentiation, and they required a large number of samples to infer regulatory relations based on statistical analysis of TF and gene co-expression (9), or they only describe one type of B cells (10). Other networks are only built with a limited set of regulators (11) or based on review of the literature. For example, (8) describes two main sets of opposed regulators during B cell differentiation. The first one inhibits the antibody producing cell identity and favors the naive B cell state, comprising BACH2, PAX5 and BCL6. The other one has the opposite effect and regroups IRF4, PRDM1 and XBP1. Most of these regulators act as inhibitors at the molecular level (8), underlining the importance to precisely characterize TF-gene relations. However, to better understand the hijacking of the normal differentiation process by follicular lymphoma, a more complete characterization of B cells transcriptional regulatory networks is required. To propose new personalized therapeutic solutions, patient-specific networks are also needed. To reach this goal, we need methods to infer regulatory networks from gene expression and regulatory regions data obtained in few biologically-close samples, able to use the system dynamics to decipher the inhibition and activation roles of regulators.

Background: regulatory networks inference Table 1 presents a survey of transcriptional regulatory networks inference methods, which were analyzed for their ability to answer our goals regarding the description of normal and pathological B cell differentiation. The criteria were: (i) uses information from gene expression, TF binding, regulatory regions and their accessibility, (ii) can work with very limited number of human samples (iii) can predict inhibitions as well as activations, i.e. provides signed networks and (iv) can be adapted to new dataset, i.e. is reproducible and reusable.

In terms of data, 11 out of the 14 methods reviewed in Table 1 use time series of gene expressions as the only input data and for 10 of them this is the only mandatory entry. This suggests that the regulatory regions impact is not taken into account in most of the methods. Few methods use information about TFs binding sites or regulatory regions - among them we can find *Regulatory Circuits* (10) - and none of them checks whether the proposed TFs are expressed in order to play their predicted roles. The resulting networks may then contain candidate relations which are not consistent with the biological situation and with limited relevance to use in a personalized medicine clinical setting.

We also observed that most methods from Table 1 produce networks with weighted edges, based on statistical or probabilistic analyses and require large datasets acquired at several time points. This required number of samples is a strong limitation to the application of these methods to human data (in particular for clinical data) with a limited number of available samples when considering disease-related and patient-specific regulatory networks. Indeed, many of these inference methods have only been tested on *Escherichia Coli* expression data and are limited to small subset of genes. Only four of them present an application to human regulatory networks, raising the question of their scalability and application to human settings.

Finally, we noticed that only the two most recent methods (23, 24) predict the activator or inhibitor role of the inferred regulations to generate signed networks. These methods specify whether the regulation is an activation or an inhibition, based on expression correlations between TFs and genes, but they ignore both TF expression levels and binding site accessibility. These issues strongly limit the possibility to use such methods to understand the biological mechanisms at play and to design targeted treatments with limited secondary

effects.

Regulatory Circuits The only method which overcomes the main limitations of the reviewed methods in terms of use of regulatory landscape information and applicability to clinical settings in human is the *Regulatory Circuits* project. This work was a great effort to merge impressive amounts of data from the ENCODE (25), FANTOM5 (26) and RoadMap Epignomic (27) consortia to describe human cell-type or disease specific regulatory networks (10). The project resulted in 394 cell type-specific gene-regulatory networks, in which TF-gene relations are associated with different weights according to the considered cell type. However, as noticed above, a single network in the dataset (named "CD20+ B cells") encompassed all the B cell subtypes of our case-study. This prevented us to understand the differences between B-cell subtypes, underlining a granularity issue.

As a first step towards the adaptation of *Regulatory Circuits* to the analysis of our case study, we successfully integrated the data used to build TF-gene regulatory networks into a unique structured graph (28), therefore facilitating the re-use of this resource. More precisely, in this previous work, we used Semantic Web Technologies, a generic data and knowledge integration framework (29, 30), to generate a unique RDF dataset that can be queried by dedicated SPARQL queries. This allowed recomputing the relations between a TF and a gene published by the *Regulatory Circuits* project. Although our approach allowed to improve the *Regulatory Circuits* project results reusability, it also highlighted that its design makes it impossible to provide information about the activating or inhibiting role of TFs involved in regulations. In addition, TF expression was not used as a selection criterion for relevant TF-gene relation. Finally, the *Regulatory Circuits* project methodology is based on computing ranks for gene expressions and regulatory regions activities. This might be suitable for a large number of samples (808 in the original project), but is not applicable to precisely describe changes between few related cell types, because i) it forces differences even between very similar values, and ii) it does not take the amplitude of expression / activity changes into account.

Objective To address the above-mentioned issues, we introduce *Regulus*, an innovative gene regulatory networks inference tool to find the regulators of gene sets with similar expression dynamics and to qualify these TF-genes' relations as either activation or inhibition.

Regulus is based on Semantic Web Technologies, extending the methods introduced in (28). Its main principles are to (1) take into account regulatory factors (TF and regions) activities, (2) propose an original discretization of the activities into patterns, (3) produce signed networks inferred by a logical consistency step and (4) be easily reusable and applicable to many datasets.

Regulus has been developed to be stringent and to limit the space of the candidates TF-genes relations highlighting the candidate relations which are the most likely to occur. By applying *Regulus* to published or original datasets, we show that it can describe regulatory networks with validated signed relations, it is able to highlight known regulators of a specific biological process and provides a list of candidate new regulators, that can be further refined. This confirms that Semantic Web technologies, which had been instrumental in the expansion of the Linked Open Data initiative (31), and in particular in life science data integration, (32, 33), are also a suitable framework for gene-regulatory network inference in the context of personalized medicine.

2. Methods

2.1. Main characteristics of *Regulus*. *Regulus* is available as a Conda / SnakeMake package at https://gitlab.com/gcollet/regulus. It allows, from gene expression and regulatory regions activity tables, with their respective genomic coordinates, to infer consistent TF-region-gene relations with patterns indications for all entities. As inputs, *Regulus* requires: (i) a list of genes with their expression in a selected number of cell types, (ii) a list of selected regulatory regions with their activity as two text files, (iii) genomic locations of the genes and regions as bed files and (iv) TF binding sites locations (bed file). These can be provided by the user (TF data from a specific ChIP-seq analysis for example), but we provide an implementation using the genome-wide TF binding sites coordinates from *Regulatory Circuits* (10), containing curated binding sites for a total of 643 TF. All genomic coordinates are given according to the *hg19* human reference genome.

Regulus outputs a list of candidate signed TF-genes relations that can be explored to identify new regulators.

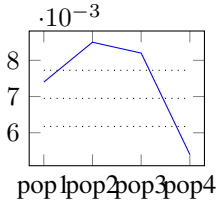
2.2. Pre-processing in *Regulus*.

2.2.1. Gene Expression and Region Density Patterns. *Regulus* principle is based on the hypothesis that sets of genes with a common expression dynamic are regulated by common factors. It also requires to define relative levels of expression or activity to perform its consistency check. Therefore, features (genes or regions) with differential activities (expression or accessibility) are defined by the user and grouped into patterns. This discretization is performed independently for each feature. A feature expression pattern is based on a several digits pattern, one for each cell population, each digit having a value ranging from 1 to 4. To determine this pattern we first compute the mean per populations based on normalized count from the differential expression analysis and log-transform it. We then use a discretization procedure on these average values where the interval between the maximal and the minimal values is divided into four equivalent intervals, providing a scale from 1 to 4. Each averaged expression value therefore gets an attribute from 1 to 4 corresponding to the interval it belongs to. Patterns are computed by a *python* script.

Figure 1 shows the pattern attribution of a gene, Figure 1a shows the expression of the gene in read count per million and the attributed pattern for each cell type. Figure 1b illustrate the pattern creation: the higher point of expression is put

	pop1	pop2	pop3	pop4
Activities	0.0074	0.0085	0.0082	0.0054
Pattern	3	4	4	1

(a) Expression of a gene (read counts per million) and expression pattern *by cell type*



(b) Graphical representation of a gene expression

Figure 1: Example of a gene expression and pattern: This gene expression is characterised by three high expression values followed by an low one. This expression is modelled by the 3441 pattern.

to 4, the lowest to 1 and then the space is divided in four equal intervals, each resting point is put to their interval number, in this case leading to the pattern 3441.
Finally, genes with low expression as defined in differential expression analysis have been granted the profile *0000* and have been removed from the implementation. Genes and regions with constant expression in all population have been granted the profile *5555*.
The TF patterns are those of their respective coding genes.

2.2.2. Neighborhood relationship. Genomic coordinates were used to compute distances between regions and genes with a custom *python* script. The distance was calculated between the two closest extremities of the entities, regardless of their respective position. All distances were filtered at a max threshold of 500 kb and set to 0 for overlapping features.

2.2.3. Finding TF binding sites in our regions. *Regulatory Circuits* (10) data on TF localization across the genome were used, as they contain reliable and extensive information. The *Bedtools intersect* (34) tool was first used to identify all the TF binding sites included into a set of regions. Then, for a given TF, only the occurrence of at least one binding site was kept, producing a binary relation between TFs and regions.

2.3. Data graph for the integration and query in *Regulus*. From the previous pre-processed data transformed from tabulated to TTL format by a custom *python* script, *Regulus* generates a structured RDF graph of data that can further be queried with Semantic Web technologies.
The construction of the RDF dataset requires the introduction of unique identifiers and of reified entities (35, 36) to describe some relations. (1) *Identifiers* For the regions, we used a unique identifier designed after the type of region (i.e. ATAC_; and Region_ in the text) followed by the row number at which they appear in the region localization file - this ensured that a same number was never used twice for different entities. For genes and TF, we kept their usual names (HGNC Gene Symbols) as identifiers. (2) *Reified entities* As RDF does not allow relations to bear a score, relations of distance between genes and regions and of TF binding into regions were both inserted by using reified relations. This method generated new entities with devoted identifiers and

```
PREFIX : <http://www.semanticweb.org/user/ontologies/2018/1#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT DISTINCT ?Gene ?Pattern_Gene ?Region
    ?Pattern_Region ?TF ?Pattern_TF
WHERE {
    ?Gene_uri :next_to_gene ?Region_Closest_uri .
    ?Region_Closest_uri :next_to_region ?Region_uri .
    ?TF_inclusion_uri :has_binding_site_in ?Region_uri .
    ?TF_inclusion_uri :binding_site_of_TF ?TF_uri .
    ?Gene_uri rdf:type :Gene .
    ?Gene_uri rdfs:label ?Gene .
    ?Gene_uri :PatternGene ?Pattern_GeneCategory .
    ?Pattern_GeneCategory rdfs:label ?Pattern_Gene .
    ?Region_Closest_uri rdf:type :Region_Closest .
    ?Region_Closest_uri rdfs:label ?Region_Closest .
    ?Region_Closest_uri :Distance ?Region_Closest_Distance .
    ?Region_uri rdf:type :Region .
    ?Region_uri rdfs:label ?Region .
    ?Region_uri :Pattern_Region ?Pattern_RegionCategory .
    ?Pattern_RegionCategory rdfs:label ?Pattern_Region .
    ?TF_inclusion_uri rdf:type :TF_inclusion_ATAC .
    ?TF_inclusion_uri rdfs:label ?TF_inclusion .
    ?TF_uri rdf:type :Transcription_Factor .
    ?TF_uri rdfs:label ?TF .
    ?TF_uri :PatternTF ?Pattern_TFCategory .
    ?Pattern_TFCategory rdfs:label ?Pattern_TF .
    FILTER ( ?Region_Closest_Distance < 500000 ) .

}
```

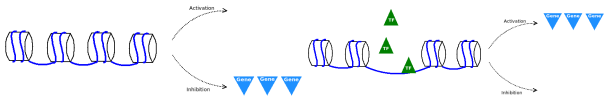
Figure 2: SPARQL query for retrieving all relations between TF-Region-Gene and their associated patterns.

TF gene	1	2	3	4	5
1	++				
2	++				
3	++	+			
4	++	+			
5	++	+			

(a) Extraction of the regulation sign attribution table (+), for an close region (pattern=1)

TF gene	1	2	3	4	5
1			-	- -	-
2		-	- -	-	-
3	-	- -	-		
4	- -	-			
5	-	-			- -

(b) Extraction of the regulation sign attribution table (-), for an open region (pattern=4)



(c) Closed chromatin and non-expressed TF: *Activation: there is a potential lack of gene expression. Inhibition: the gene could be absent.*

(d) Opened chromatin and expressed TF:*Activation: there is a potential high gene expression. Inhibition: the gene expression could be high.*

Figure 3: Illustration of the regulation sign attribution table.

bearing either the distance value or the inclusion. (3) *Additional information* References to Uniprot and Ensembl were added for the genes to link our data with public databases. We also kept the localization information for all features, as a potential filtering parameter.
To find the potential regulators of a given set of genes, *Regulus* uses the SPARQL query described in Figure 2 on the previously generated graph of data to extract all TF-Region-Genes triples and their related patterns.

2.4. Consistency table to assign roles to relations in *Regulus*. *Regulus* relies on a principle of consistency between genomic landscape, genes and TF expressions to decide if a relation is susceptible to exist. This consistency also allows predicting an inhibition or activation effect for each relation. Figure 3 bottom presents a graphical representation of

the consistency principle behind assignment tables. When the chromatin is not accessible and the TF not expressed (First line of Figure 3a): if there is no gene expression then the TF is more likely to act as an activator and if there is a high gene expression then the TF is more likely to act as an inhibitor. When the chromatin is accessible and the TF expressed (Figure 3b): if there is a gene expression then the TF is more likely to act as an activator and if the gene expression is low or null, then the TF is more likely to be an inhibitor.
These principles are illustrated in Figure 3 top: the window of activation (respectively inhibition) glides to lower (higher) gene expression as the TF expression decreases. The same behavior is true when the region accessibility decreases. The only exception to this is when the value of one or more feature is set to 5, which means it is constant across all studied populations: then, the output of the TF-region-gene ternary relation depends only upon the remaining variable elements. For each digit (corresponding to a specific cellular population) of the three patterns (gene, TF and region), the consistency of the relation with an activation or an inhibition is screened by *Regulus*. If it is the case, a score of 1 or 2 is given to the digit depending on the confidence ("-" and "+" award a score of 1; "- -" and "++" award a score of 2), and the next digit is considered. The sum all of points must be superior to a fixed threshold to award the relation a sign, either + or - depending of the direction. For example this threshold is fixed to 7 for a 4 digit pattern, allowing at most one digit to be of lower confidence (see Figure 3).

2.5. Finding key regulators with *ClassFactorY*.

Regulus output tables can be merged to obtain unique TF-gene relations and refined by using the coverage / specificity and GO / MeSH annotation scripts available as the standalone external tool *ClassFactorY* at https://gitlab.com/EveBarre/ClassFactorY.

2.5.1. Coverage and specificity filters. The first filter applied by *ClassFactorY* is on coverage and specificity of TF for some gene patterns. (a) *Definitions and computing* Coverage of a TF is calculated as the proportion of genes in a specific pattern which are targets of a given TF. The coverage itself does not provide enough information: the smaller patterns, sometime composed of 1 or 2 genes, are easily fully covered by a TF. Specificity is based on the proportion of targets genes that are from a specific pattern, for a given TF . A TF has a great specificity for a pattern if out of all its target a significant number comes from this pattern. As for the coverage, the specificity does not bring enough information by itself: despite having a large number of its targets in a pattern, a TF may have little influence on it if the pattern itself is very large. Both the coverage and the specificity are calculated as percentages. (b) *Combination of coverage and specificity* For a given pattern, a TF of interest is a TF which specificity and coverage are both superior to a threshold chosen by the user: mean + one standard deviation, quantiles or specific percentages. Then *ClassFactorY* outputs a list of selected TFs together with the gene patterns they potentially regulate.

2.5.2. GO and MeSH annotation. To validate the TFs inferred by *Regulus*, the stand-alone module *ClassFactorY* allows for automatic queries of the GO, Uniprot and PubMed databases. A user-defined list of GO annotations is used to verify if candidates TFs are annotated by these terms. The MeSH terms are retrieved from a user-defined list of PubMed publications about the biological context and the module counts the number of citations associating the TFs to one or several of the terms. An annotation-based score is then calculated and provided as a help-decision tool for end users.

2.6. Datasets used for validation and testing.

2.6.1. "Roadmap Epigenomics" RNA-seq datasets. For validating gene inclusion in the networks computed by *Regulus* we used the same "Roadmap Epigenomics" RNA-seq datasets and methods that were used in (10). Basically, RNA-seq datasets from Gtex corresponding to the *FANTOM5* tissues used for network inference were separated in three gene sets corresponding to the 10% most expressed ones, 10% less expressed ones and the 10% in the center of the expression distribution. For each category, we then checked if the genes were included in the inferred networks and gave the percentage of retrieved genes.

2.6.2. B-cells datasets. To investigate *Regulus* prediction abilities, we used datasets of differentiating B cells (accessible on demand), comprising 3 replicates of RNA-seq for naive B cells (NBC), IgM secreting or IgG secreting memory B cells (IgM+ or IgG+ MBC). Data were aligned on the hg19 human reference genome and gene expressions for 26,734 genes were calculated with *featureCounts*. Raw counts files were then used for differential expression analysis with *DESeq2*, allowing us to create a list of unexpressed (0000 pattern), invariable (5555 pattern) and variable genes and TF (used to compute relations). ATAC-seq data was obtained on the same cell types (n = 1), aligned on hg19, 35,078 regions were called with MACS2. An union of all regions was made and reads were counted for each region, normalized by sequencing depth and region size to compute read densities, using an in-house *bash* script. Densities were then used to create patterns as described above.

3. Results and Applications

3.1. The *Regulus* tool.

We designed *Regulus*, a transcriptional regulatory networks inference tool dedicated to the analysis of few and biologically-close datasets. The tool relies on Semantic web to integrate expression and epigenetic data. Figure 4 presents the different steps of the *Regulus* pipeline. The preprocessing steps (blue steps in the Figure Figure 4d, detailed in Figure 4a) consist in (i) transforming individual gene expressions and regulatory regions activity (read densities) into patterns, (ii) finding relations between genes and regions by computing the distances between neighboring genes and regions, (iii) finding the inclusion of TF binding sites into regions. The pre-processed data are then integrated using Semantic Web technologies (yellow step of

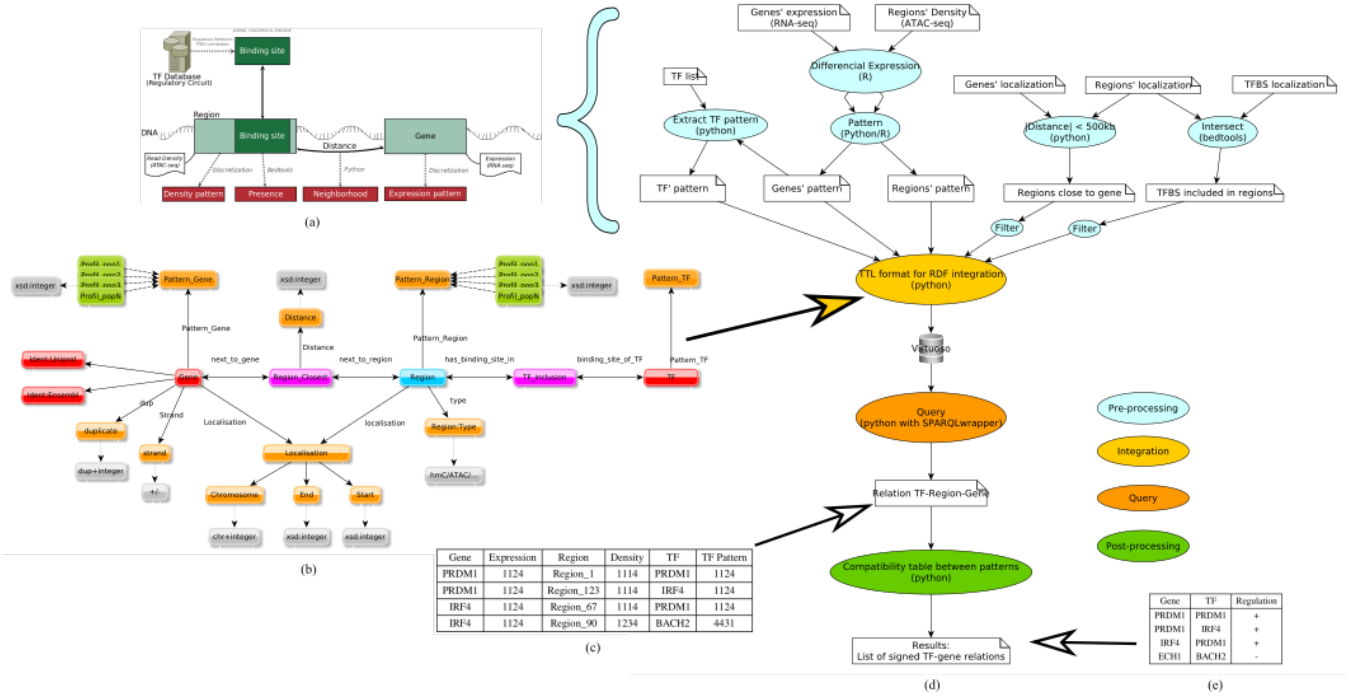


Figure 4: **Representation of the different steps of the *Regulus* pipeline.** In blue: pre-processing steps based on biology (a), expliciting the relations between the entities: inclusion of the TFBS in region, distance between region and genes and in red the pre-treatment done: discretization and filtration. In yellow: integration, creation of a RDF graph (b) formalizing the relation between all the entities and their attributes: the relations between TFs and genes can be found by following the edges of the knowledge graph. In orange: query (see Figure 2) to extract all the relations following the requirement explicited by *Regulus*, the result in this step is an overall unsigned regulatory network (c) expliciting relations as TF-Region-gene triples and their respective patterns. And in green: the application of the compatibility table (see Figure 3) resulting of a signed and filtered regulatory network (e).

Figure 4d, producing a typed RDF graph whose relations between entity types are shown in Figure 4b). They can then be queried to identify the candidate relations between TF, regions and genes that match a given set of rules: the regions must be at most at 500 kb of the gene, the TF must be expressed and have a binding site in the region (orange step of Figure 4d). The output of this step is a gene-region-TF interactions table (see Figure 4c) Query results must then be refined (green step of Figure 4d) to identify the TF-gene relations that are compatible with their expressions and assign them signs (either positive, indicating an activation, or negative, indicating an inhibition). The final output of the process is a signed TF-gene interaction table (see Figure 4e).

Pre-processing data for efficient integration: patterns as descriptions of data variability The preprocessing step of *Regulus* allows to aggregate information according to the concept of patterns (see Section 2.2.1 for details). The relevant features (genes and regulatory regions) for building a regulatory network are those which vary between the compared cell types or populations. Computed patterns therefore regroup features which exhibit similar expression variations between cell populations regardless of their respective absolute expression levels. By construction, the patterns do not show a chronological order between the different populations, but only describe the expression dynamics. The network inference method is therefore based on the common assumption that genes sharing a common pattern are regulated by a common set of regulators (37).

Data integration in a structure that can be browsed and queried The discretized patterns are integrated into an RDF graph by *Regulus*. To do so a first step was to ensure that all the necessary files were formatted for the integration. The data model structure after integration is illustrated in Figure 4b, which can be seen as a representation of the interactions between the data, where the entities are linked between each other by explicit relationships. To retrieve TF-Gene relations, all the entities presented in Figure 4b are not necessary, some of them have been added to help refine the results. The entities that are strictly necessary are: genes, TF and regions with their respective patterns, as well as the reified entities Region_closest and TF_inclusion.

From the data structure we generated the query starting from the node "*Gene*" having an expression pattern, we trace back all *Region* which are connected to the *Gene* by a *Region_closest* relationship. We filter these *Regions* by taking into account the *TF* which have a relationship with the *Region* through *TF_Inclusion*, representing the presence of a binding site. Along the way, we gather the patterns for the TF, the region and the gene, as we will need them in the next step. This lead to the SPARQL query presented in Section 2.3.

A logical consistency step for filtering and sign attribution To discard the TF-gene candidate relations that are not consistent with the biological knowledge of how regulation works, and to infer a regulation sign (i.e. activation or inhibition) for the consistent candidate relations, we used a consistency principle.

It is based on knowledge about the following biological prin-

ciples for gene regulation: **(i)** the maximum effects on the gene expressions are obtained when the TF is at its highest expression level. **(ii)** The more accessible the region, the higher is the impact of the TF on the gene: for an activation this implies that if the TF is highly expressed so must be the gene, for an inhibition the higher the TF, the lower the gene's expression. **(iii)** If the region looses its accessibility, the weight of the TF lowers: a higher TF expression level is needed to get a similar effect on gene expression. **(iv)**For fine grained tuning, we used two levels of confidence to weight the consistency between the feature patterns and the relation sign.

These principles were transcribed into rules and used to filter and sign the consistent relations (see Section 2.4 for details about basic principles, scoring and thresholds). After using the consistency table for filtering we obtained a result in form of a quadruple between the gene, the neighboring regulatory region, the TF with a binding site in the region and the signed potential regulation on the gene such as presented in Figure 4. These relations can be merged into unique TF-gene relations when consistent relations involving the same TF and gene are found through different regions.

3.2. Application to *FANTOM5* data: validation of the basic principles.

One aim of *Regulus* is to compute regulatory network on a limited number of samples and cell populations. We therefore chose to run *Regulus* on four limited datasets extracted from *FANTOM5*, each containing four tissues (= cell population), chosen to be either similar in origin (comparable organ) or dissimilar (widely different localization). Detail of the chosen subset is in Table 5a.

3.2.1. Regulus generates large networks which includes low expressed genes.

Networks topology is refined by the consistency step. Table 5b presents the number of relations obtained by *Regulus* for these four datasets. Data comprise 16,888 genes expressions, 43,012 regulatory region activities and 124,358,159 TF binding sites genomic coordinates. Just after the query and before running the consistency table, we can see that we had the same number of relations (3,005,934 TF-region-Gene or 1,869,854 TF-Gene), this is due to the fact that we have the exact same information on the TF binding sites and regions for the four sets.

As seen in Table 5b the consistency step allows to generate networks which are different in size and quality. They represent about 10% of the possible relations predicted by the query, underlining the filtering power of the consistency step. Networks computed on dissimilar sets of tissues are slightly smaller than the ones computed with similar subsets. This lower number of relations may be explained by the increased lack of consistency in TF-gene regulations across widely different tissues.

Inclusion of low-expressed genes. For validating gene inclusion of our networks we used the same "Roadmap Epigenomics" RNA-seq datasets that were used in (10). As seen in Table 5c, for the networks computed using *Regulus*, we recovered highly and medially expressed genes at high levels (> 90% for each category). Low-expressed genes were also significantly included in our networks, with a retrieval rate ranging from 36.75% for similar cell-types to 52% for dissimilar subsets.

Sub-set Name	Tissue 1	Tissue 2	Tissue 3	Tissue 4
Dataset 1	B lymphoblastoid cell line	CD4+ T cells	CD8+ T cells	peripheral blood mononuclear cell
Dataset 2	colon adult	colon fetal	small intestine adult	small intestine fetal
Dataset 3	CD34+ stem cells adult	brain fetal	epitheloid cancer cell line	pancreas adult
Dataset 4	CD4+ T cells	brain fetal	colon adult	epitheloid cancer cell line

(a) Composition of the tissues of the 4 sub-sets from *Fantom5*: 2 composed of biologically-similar tissues and 2 composed of dissimilar tissues.

Sub-set	Nb relations TF-region-Gene Predicted before filtering	Nb relations TF-region-Gene After filtering	Nb relations TF-Gene After filtering	Nb relations TF-Gene "+"	Nb relations TF-Gene "-"	Ratio relations TF-Gene +/-
Dataset 1	3,005,934	219,495	164,251	114,624	49,627	2.3
Dataset 2	3,005,934	237,487	178,514	154,276	24,238	6.3
Dataset 3	3,005,934	165,804	125,451	79,359	46,092	1.7
Dataset 4	3,005,934	165,597	126,145	88,609	37,536	2.3

(b) Number of relations by network after *Regulus*, on the 4 sub-sets of *Fantom5*.

Genes regulated in	Top 10% most expressed	Mid 10%	10% least expressed
Dataset 1	90.75	86.25	41
Dataset 2	90	88.25	32.5
Dataset 3	89	88	49.25
Dataset 4	88.75	85.75	55

(c) Percentage of genes from the RNA-seq related to the tissues found in the resulting networks. The RNA-seq genes are separated in three categories: the top 10% most expressed, the middle 10% and the 10%least expressed.

Relations	Trrust				Signor				Trrust ∩ Signor			
TF-gene in	Total	True	False	Unknown	Total	True	False	Unknown	Total	Different	True	False
Dataset 1	432	164	65	203	76	54	21	0	51	9	33	9
Dataset 2	357	156	37	164	54	45	9	0	39	5	31	3
Dataset 3	293	100	54	139	56	42	14	0	39	4	28	7
Dataset 4	264	113	50	101	58	44	14	0	37	2	26	7

(d) Relations found in Trrust and Signor and coherence of signs. True: number of relations with the same sign as the database, False: relations with different sign than the database, Unknown: relations non signed or signed + and - in the database. For the union of Trrust and Signor: different: relations signed differently in the two databases, True: relation signed the same as both databases and False: relations signed differently than the databases.

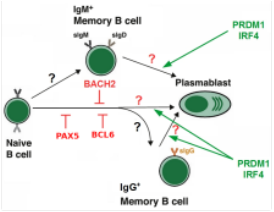
Figure 5: Statistics on the regulatory network computed on four datasets extracted from *Fantom5*: list of tissues composing the datasets, their number of interactions, their validations with public databases and validation with the RNA-seq.

3.2.2. Inferred relations signs are validated by public data.

To validate the signed relations inferred for the networks using *FANTOM5* data, we looked at the two major databases containing signed regulatory relations: Trrust (38) and Signor (39). Trrust and Signor both contain signed relations that have been described in the literature. Some of these relations can be unsigned or contradictory signed (within the same resource or between both), as evidence for activation or inhibition depends on the biological context. Trrust contains 9,396 relations (4,325 of which with unknown signs) of those only 2,428 can appear in our data meaning both TF and Genes present in our dataset. Signor contains 2,820 relations and only 851 we can obtain.[1]

In Table 5d we compiled the relations in our networks that are found in either Trrust or Signor. In average we found 0.22% of the computed relations of a set in Trrust and 0.04% in Signor. In Trrust 45% of the relations are found unsigned, 39% signed in the same direction and 16% signed differently as the database. In Signor 76% are signed the same way and 23% in opposite direction.

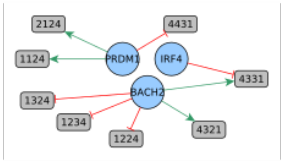
[1] Signor and Trrust were queried the 28 April 2020.



(a)

Class	Data types	Number of entities
Gene	Start, End, Duplicates, Transcripts, Chromosome, Strand, Expression Pattern	26,802
ID ATAC	Start, End, Chromosom, Density Pattern	35,090
ID ATAC Closest	Distance to gene, Neighbor Gene, ID ATAC	496,282
TF	Expression Pattern	593
TF inclusion ATAC	ID ATAC, TF	3,460,101

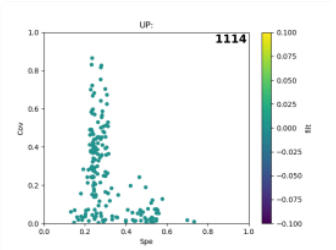
(b)

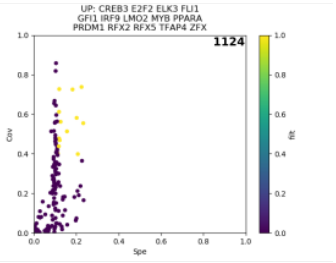


(c)

Pattern or nb Patterns	Nb of genes in pattern	Nb genes (Union)	% total genes	Nb TFs in pattern	Nb TF-genes relations	% total relations	% of relations "+"	TF targeting the pattern	nb TF passing filter (cov/spe)	% TF kept	Biological interpretation
109 (ALL)	26,734	26,734	100 %	593	314,965	100 %	55.15 %	266	121	45.49 %	
1 (0000)	14,921	14,921	55.81 %	327	0	0 %	0 %	0	0	0 %	Not expressed
1 (5555)	3,591	3,591	13.43 %	51	73,505	23.34 %	71.53 %	42	0	0 %	Not variations
1 (4441)	1,418	1,418	5.30 %	62	66,464	21.10 %	64.04 %	186	0	0 %	Decreasing expression in PB
1 (1114)	1,244	1,244	4.65 %	11	58,769	18.66 %	37.89 %	184	0	0 %	Increasing expression in PB
16	[102-557] mean: 231 median: 173	3,704	[0.38-2.08]% mean: 0.87% median: 0.65 sum: 13.86%	[0-15] mean: 5 median: 3.5	[618–19,833] mean: 6,388 median: 2,986 sum: 102,204	[0.2-6.3]% mean: 2.03% median: 0.95% sum: 32.45%	[18.9-81]% mean: 47.9% median: 43.3%	[51-147] mean: 116 median: 120	[10-23] mean: 16 median: 15.5	[8.2-20.9]% mean: 14.8% median: 15.8%	Potentially interesting: nb genes
92	[[1-76] mean: 21 median: 13	1,856	[0-0.28]% mean: 0.08% median: 0.05% sum: 6.94%	[0-4] mean: 0.72 median: 0	[0–1,156] mean: 157 median: 46 sum: 14,023	[0-0.37]% mean: 0.05% median: 0.01% sum: 4.45%	[0-100]% mean: 51.3% median: 50.7%	[0-107] mean: 32 median: 21	[0-17] mean: 4.6 median: 2	[0-33.3]% mean: 11.2% median: 12.5 %	Difficult to interpret: low nb of genes

(d)





(e)

Figure 6: **Synthesis on the B cell regulatory network.** (a) Biological interaction between the cell types and main known regulators of the B cell differentiation process. (b) Class name, data type and number of entities integrated in the RDF/SPARQL endpoint. (c) Graph of interaction of the main known regulators and their targeted patterns after filtering with coverage and specificity: IRF4 & PRDM1 are described as activators of the PB identity and BACH2 is given as an inhibitor of the PB differentiation. (d) Summary on all the patterns for a selection of properties. Only 19 patterns are composed of more than 100 genes and they represent 95.55% for the overall relations. Of the total number (266) of regulators in the regulatory networks only 121 pass the coverage and specificity threshold meaning that 45.5% of the TFs are keep, but in average, looking by patterns only 15% of their regulator are selected. (e) Distribution of the coverage and specificity for all the TF targeting 1114 (top) and 1124 (bottom): yellow dots indicate the TFs passing the threshold. While being one of the largest pattern 1114 does not have any key regulator with our filtering method.

The relations found in common between Trrust and Signor in our networks are signed the same way in both resources in most cases and in 72% signed the same way as in our networks. For the relations generated by *Regulus* and found at least in one of Signor or Trrust, the sign we predicted is also consistent with the databases in two third of the cases. It is important to note that Trrust and Signor relations are not necessary found in the same tissues as the ones we used, which may explain some of the differences in signs.

3.3. Application to B-cells identifies known and potential new key regulators.

After having determined that *Regulus* performed well on closely related cell populations with few samples, we decided to test it on some new unpublished data generated in our group, about B cell differentiation.

3.3.1. Biological context. After stimulation by a pathogen, naive B cells (NBC) differentiate into either memory B cells (MBC) or plasmablasts (PB). MBCs store some information about pathogen encounter and are able to differentiate faster and more efficiently if the same pathogen is present again. PB are effector cells and produce antibodies to inactivate and eliminate the foreign pathogens. We wanted to probe regulatory networks to explain the different abilities of NBC and MBC to differentiate into effector cells. For this study, we used four distinct populations: NBC, IgM$^+$ MBC (MBC IgM) , IgG$^+$ MBC (MBC IgG) and PB. We generated gene expression data (RNA-seq, 26,802 genes) and chromatin accessibility data (ATAC-seq, 35,090 regions) that can be used to determine regulatory regions activities. In this specific case the four populations are sequential: NBC is the first population and PB is the last, but MBC can be either a transitional state or a final one. Three main TF are highlighted in the bibliography at different steps of the differentiation: an inhibitor, BACH2, and two activators, IRF4 and PRDM1 (8), as shown in Figure 6a.

3.3.2. Patterns are indicative of expression dynamics. With the input data we obtained 109 distinct patterns of 4 digits representing in order: NBC, MBC IgM, MBC IgG and PB. The patterns are comprising from 1 gene (1412 and 2414) to 1,418 (4441), 18 patterns are composed of more than 100 genes (Figure 6e). These patterns indicate the main dynamics at work in our system: the most numerous 4441 pattern shows that many genes are down-regulated when either NBC or MBC are driven towards differentiation into PB. The 0000 (unexpressed) and 5555 (not variable) patterns represent 14,921 and 3,591 genes, respectively - those two patterns were put aside during the interpretation of *Regulus* results. Included within the genes but treated as a different entity, we also retrieve 593 TF from the 643 for which binding site information is available in *Regulatory Circuits*. 327 TF had a pattern of 0000 and hence would not impact the networks, leaving a set of 266 potential regulators in our system. Therefore, pre-processing the data into patterns already provides some filtering and allows the user to concentrate on patterns relevant to the biological context.

3.3.3. Potential relations need to be consistency-filtered. Pre-processed data were then integrated to create the data structure and the relation graph (Figure 6b). Once integrated and queried, *Regulus* output from our data is a set of 5,635,099 TF-regions-genes relations, which resulted in 612,633 TF-region-gene signed relations, once filtered with

the consistency table. This number is further reduced to 314,965 unique TF-gene relations, by merging TF-gene relations that happen through different regions. Of those relations 173,717 are signed as activation (+) and 141,248 as inhibition (-).
For the 18 most numerous gene expression patterns, the number of unique TF-gene relations ranges from 618 to 66,464, involving 51 to 186 TFs and 102 to 1,418 genes. The number of relations by gene ranges from 6 to 47 (median at 24) and the percentage of activation in the relations ranges form 15% to 82% (median at 43%)

3.3.4. Key regulators can be identified by coverage and specificity filters. After filtering the resulting TFs with the specificity and coverage threshold (used q75, see 2.5.1), we look at the number of TF passing the filter (Figure 6c). For 25 patterns we highlight no specific regulator and for 16 only one TF appears to pass both threshold, finally 35 patterns have 10 or more regulators of interest. Out of the 237 TF in the resulting networks, 116 do not pass the threshold for any patterns, 23 pass in only 1 pattern and 19 are up in 10 or more patterns. The combination of both parameters allow us to filter out some highly ubiquitous TF, such as SP4.
Among the 121 TF passing the threshold, many have already been described as implicated in B cell differentiation, including the main known regulators PRDM1, IRF4 and BACH2 (8). At the pattern level (Figure 6d), IRF4 is found as an inhibitor of 4331, PRDM1 is an activator of 1124 & 2124 and an inhibitor of 4431, while the literature describe them as activators of the PB identity. BACH2 is activator of 4321 & 4331 and inhibitor of 1224 & 1234 & 1324 and is given as an inhibitor of the PB differentiation. *Regulus* results are therefore in agreement with the literature, since in our patterns, PB is represented by the last digit of our patterns.
Finally, we provide some metrics about TF-gene relations based on the number of genes in each pattern (Figure 6e). We observe that for the two non-constant most populated patterns (1114 and 4441), no TF could be singularized by the coverage / specificity filter (Figure 6c). Importantly the 19 patterns counting more than 100 genes more than 95% of the relations (more than 72% if constant genes are excluded), while less numerous patterns (n=92) only regrouped a small fraction of all TF-gene relations.

3.3.5. Regulators annotation as a decision helping step. The two annotation steps (see 2.5) allow for validation of the already known TFs, relevant in the biological context (For the B cells we chose the following GOterms: cellular developmental process, immune system process, lymphocyte activation, B cell activation, plasma cell differentiation & B cell differentiation) and provide a list of potential TFs of interest - not described as regulators in the given context - for further biological experiment.
Out of the 121 TFs identified in the previous step: 64 were annotated by cellular developmental process, 27 by immune system process, 13 by lymphocyte activation, 3 (IRF8 & LEF1 & YY1) by B cell activation, 2 (IRF8 & YY1) by B cell differentiation and none by plasma cell differentiation.
We also looked at the number of citation in Pubmed for the found TFs, ranging from 164,841 (MAX) to 2 (ZNF75A). The number of publication for the TF and the MeSH terms: *(Plasma Cells OR B-Lymphocytes OR Lymphocyte Activation OR Germinal Center) AND (Transcription Factors OR Gene Expression Regulation) AND Cell Differentiation* allow to verify the specific context of the B differentiation. 93 TFs are cited with those MeSH terms amongst which IRF4, STAT3, MYC, PRDM1 & PAX5 being the 5 TFs with the most citations (208 to 386). The second query was done with the TFs and the MeSh terms: *B-Lymphocytes OR Plasma Cells*, giving a more global context to the B cells. 104 TFs are annotated in this case: PRDM1, PAX5, STAT3, MYC & MAX being the 5 with the most citations (402 to 2573). Interestingly, a null score was attributed to six TFs (KLF16, FOXJ3, TFAP4, TGIF1, ZNF219 and ZNF75A). This means that although they have been selected by the coverage / specificity filter, their potential role in B cell has not been studied and may be worth investigated.

3.4. *Regulus* compares favorably to *Regulatory Circuits*. After validating our pipeline, we wanted to compare it to the closest method of network inference in terms of dataset size and nature of the genomic features, *Regulatory Circuits*.

3.4.1. Workflows comparison. Figure 7b presents the main steps of the *Regulus* pipeline, and compares them to those of *Regulatory Circuits* (Figure 7a). Both take as input: activities of the regions and the genes (or approximation of the transcript activity) and regions localization. As shown in Figure 7, both also share similar pre-processing steps like computing the distance between regions and genes or finding the TF binding sites occurrences in the regulatory regions. The main differences are:

- Our method uses the activity of the TF (extracted from its coding gene activity) which was not taken into account by *Regulatory Circuits* workflow.
- *Regulatory Circuits* uses a composite score in which each component must be strictly positive, and is taken as a maximum when several concordant relations exist. This approach brings a bias towards activation relations and favors highly expressed genes. On the contrary, our method checks the consistency between the different activities of the relation entities to produce signed networks.
- *Regulatory Circuits* gives tissue-specific networks whereas *Regulus* outputs networks by patterns, adding dynamics to the network.

3.4.2. Networks and outputs comparison.

Networks topology. For the *FANTOM5* data used in 3.2, *Regulatory Circuits* number of potential TF-genes relations (2,060,960) was calculated based on the supplementary data files describing entities (TF, promoter, enhancer, transcript) relations, while ignoring the scores (see (10) for details and

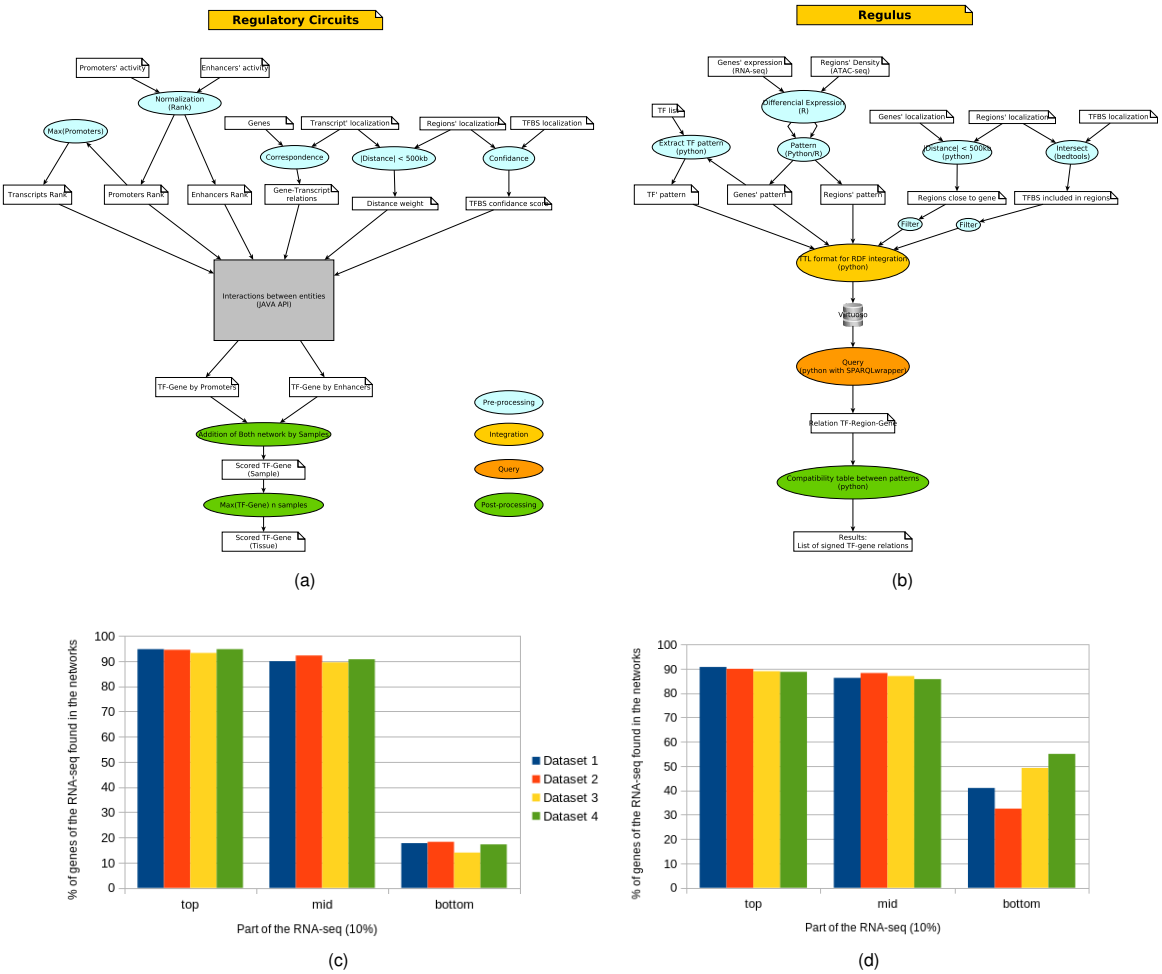


Figure 7: **Comparison of the *Regulatory Circuits* workflow (a) and *Regulus* (b):** the pre-processing (in blue) steps are similar: normalization of the expressions, limitation of the distance between region and genes and inclusion of the TFBS in the regions. The main differences are: *Regulus* use RNA-seq of the genes activities where (a) use the activity of their promoters & (b) use a clusterization by pattern while (a) use a rank normalization without clusterization. While the construction of the regulatory network use different technologies (gray, yellow and orange) they follow similar principles: finding the biological relations between the entities and retrieving the patterns (b) or weight (a) along the relations. The main differences are in the post-processing (green): (a) gives a score for each relation in a sample and filter the score > 0 and then compute the tissues networks, while (b) compute a regulatory network and filter it with a consistency rule and sign every interaction. In (c) using *Regulatory Circuits* and (d) using *Regulus*, we look at the percentage of genes from the RNA-seq related to the tissues found in the resulting networks, the datasets are the one presented in section 3.2 and fig 5. Overall *Regulus* is better at recalling the least expressed genes (from 15% to 35%) and similar on the other points.

data availability at [2]). All the TF-gene relations found by *Regulus* are included in the potential *Regulatory Circuits* relations, meaning that our method does not create irrelevant relations.

However, *Regulatory Circuits* provides very large networks, comprising from 407,056 to 1,796,098 unsigned relations for a single tissue or cell type; whereas *Regulus* signs and refines these relations to less than 180,000 for sets of four different tissues, adding an interesting filtering power.

Inclusion of genes based on their expression level. When using a similar validation of output genes using Gtex RNA-seq datasets as in 3.2, *Regulatory Circuits* was well able to retrieve highly or moderately expressed genes in its networks. However, as shown in Figure 7c, lowly expressed genes were poorly incorporated, in agreement with their methodology favoring inductive relations and high activities (see above). Whereas *Regulus* manages to retrieve similar percentage of highly and moderately expressed genes but higher percentage of lowly expressed ones, see Figure 7d.

[2] http://regulatorycircuits.org/

4. Discussion

In this article we presented a new design for regulatory network inference. *Regulus* addresses some of the methodological issues presented in Introduction: (1) the under-exploitation of the regulatory context, (2) data reduction and structure, (3) the lack of functionality qualifier for interactions (activation of inhibition) and (4) pipeline availability for reuse and reproducibility. To solve theses issues (1) we added the TFs expression and regulatory regions activities in the pipeline, (2) we chose to use patterns of expression which cluster genes or regions of similar expression trend under the same pattern and (3) to integrate them in a Semantic Web Technologies based structure, that can be browsed and queried. We computed a global network based on all the sam-

ples and then checked the consistency of the relations with the biological background, which allowed us to sign the relations and to not discriminate the inhibition. (4) We also provided an automated version of *Regulus* to facilitate its reuse. Altogether, we found that *Regulus* compared favorably to the only similar method incorporating knowledge about regulatory regions, *Regulatory Circuits*.

The solution we chose to implement to cluster genes of similar expression direction in our populations was to group them by patterns. Describing the expression or activities as patterns allows the user to concentrate on patterns which have a biological meaning in the given experimental setting. Cell types are not individualized by these patterns, but biologists are often more interested in dynamic changes between cellular states than by a complete record of regulators active in a fixed cell population. Even if they are interested in such characterization, it is possible to look at regulators for the pattern where expression is at its highest only in the cell population of interest. A limit of this approach might be the oversampling of the patterns as some patterns were very poorly populated: 33 with less that 10 genes including 18 with less than 5 genes on the B cells data. Those patterns could be grouped with patterns of similar direction or removed from the analysis, since they may bias the coverage and specificity filter. Indeed small patterns are easily covered and may introduce high specificity percentages. Another solution would have been to use co-expression analysis for example using the WGCNA R (40) package. Unfortunately, after testing, it gave poor results with our limited number of datasets.

Our work shows the added value of integrating large and heterogeneous data from biological experiments when inferring regulatory networks. Even on closely related cell types, such as the B cells subsets, we are able to identify subtle differences based on our use of patterns and of the consistency step. The Semantic Web Technologies framework allows for an easy identification of relations. Once the data structure is obtained, it can be queried to answer any specific question. As it is based on unique identifiers and self-structuring, it also reduces the risks of introducing false relations, which may happen when manipulating text files with command line tools.

We provide the *ClassFactorY* tool to identify the key regulators by introducing a filter of coverage and specificity, and by adding biological context annotations. From the 237 potential TF involved in B cell networks, the coverage and specificity reduces this number to 121, which is still too much to perform experimental validation. Annotations can thereafter be used on these "short-listed" TF to (1) validate known regulators and (2) identify potential new regulators which have not been described in the context of interest. There is still room for improvement for the reduction of the candidate regulators: a perspective is to use constraint programming to determine the smallest group of TF able to regulate the biggest part of a gene pattern.

Finally, the combination of *Regulus* and *ClassFactorY* was able to retrieve the main regulators of the B cell differentiation process, such as PRDM1, IRF4, BACH2 and PAX5 and to pinpoint them with a high annotation score. On the other hand, six new potential TFs impacting on this process, identified through high coverage and specificity coupled to a null annotation score, would need to be further investigated, showing the power and interest of our tool.

DATA AVAILABILITY STATEMENT

The data underlying this article are available in the article and in its online supplementary material. More precisely, *Regulus* software and source code are available on gitlab at: https://gitlab.com/gcollet/regulus. *ClassFactorY* software and source code are available on gitlab at: https://gitlab.com/EveBarre/ClassFactorY. FANTOM5 and GTex data used for testing and validation are available in the supplementary data of (10) at http://regulatorycircuits.org. B cells datasets will be shared on reasonable request to the corresponding author.

AUTHOR CONTRIBUTIONS

ML designed the workflow and the outline of the study. She coded the initial version. She designed the application to B cells and coded the initial version. **GC** automated and optimized the workflow. **EB** automated the validation, designed and developed *ClassFactorY* and ran the experiments on B cells. **TF** participated to the clinical expertise. **OD** participated to the design of the workflow and the study. **AS** participated to the design of the workflow and the study. **FC** contributed to the design of the workflow and the study, designed the expression patterns and designed the signed compatibility table.

ACKNOWLEDGEMENTS

We acknowledge the GenOuest bioinformatics core facility (https://www.genouest.org) for providing the computing infrastructure.

FUNDING

ML was financed by the "Médecine Numérique" joint PhD program from INRIA & INSERM. Data acquisition on B cells subsets was funded by an internal grant from the Hematology Laboratory, Pôle de Biologie, Centre Hospitalier Universitaire de Rennes, Rennes, France.

5. Bibliography

1. Cathie Garnis, Timon PH Buys, and Wan L Lam. Genetic alteration and gene expression modulation during cancer progression. *Molecular Cancer*, 3(1):9, 2004.
2. Andrea Smallwood and Bing Ren. Genome organization and long-range regulation of gene expression by enhancers. *Current opinion in cell biology*, 25(3):387–394, 2013.
3. Geeta J Narlikar, Hua-Ying Fan, and Robert E Kingston. Cooperation between complexes that regulate chromatin structure and transcription. *Cell*, 108(4):475–487, 2002.
4. Z. Duren, X. Chen, R. Jiang, Y. Wang, and W. H. Wong. Modeling gene regulation from paired expression and chromatin accessibility data. *Proc Natl Acad Sci U S A*, 114(25): E4914–E4923, 06 2017.
5. A. R. Sonawane, J. Platig, M. Fagny, C. Y. Chen, J. N. Paulson, C. M. Lopes-Ramos, D. L. DeMeo, J. Quackenbush, K. Glass, and M. L. Kuijjer. Understanding Tissue-Specific Gene Regulation. *Cell Rep*, 21(4):1077–1088, Oct 2017.
6. M. Ota, Y. Nagafuchi, H. Hatano, K. Ishigaki, C. Terao, Y. Takeshima, H. Yanaoka, S. Kobayashi, M. Okubo, H. Shirai, Y. Sugimori, J. Maeda, M. Nakano, S. Yamada, R. Yoshida, H. Tsuchiya, Y. Tsuchida, S. Akizuki, H. Yoshifuji, K. Ohmura, T. Mimori, K. Yoshida, D. Kurosaka, M. Okada, K. Setoguchi, H. Kaneko, N. Ban, N. Yabuki, K. Matsuki, H. Mutoh, S. Oyama, M. Okazaki, H. Tsunoda, Y. Iwasaki, S. Sumitomo, H. Shoda, Y. Kochi, Y. Okada, K. Yamamoto, T. Okamura, and K. Fujio. Dynamic landscape of immune cell-specific gene regulation in immune-mediated diseases. *Cell*, 184(11):3006–3021, May 2021.
7. Ekta Khurana, Yao Fu, Dimple Chakravarty, Francesca Demichelis, Mark A Rubin, and Mark Gerstein. Role of non-coding sequence variants in cancer. *Nature Reviews Genetics*, 17 (2):93, 2016.
8. Simon N Willis and Stephen L Nutt. New players in the gene regulatory network controlling late b cell differentiation. *Current Opinion in Immunology*, 58:68–74, 2019.
9. K. Basso, A. A. Margolin, G. Stolovitzky, U. Klein, R. Dalla-Favera, and A. Califano. Reverse engineering of regulatory networks in human B cells. *Nat Genet*, 37(4):382–390, Apr 2005.
10. Daniel Marbach, David Lamparter, Gerald Quon, Manolis Kellis, Zoltán Kutalik, and Sven Bergmann. Tissue-specific regulatory circuits reveal variable modular perturbations across complex diseases. *Nature methods*, 13(4):366, 2016.
11. A. Méndez and L. Mendoza. A Network Model to Describe the Terminal Differentiation of B Cells. *PLoS Comput Biol*, 12(1):e1004696, Jan 2016.
12. Shoudan Liang, Stefanie Fuhrman, Roland Somogyi, et al. Reveal, a general reverse engineering algorithm for inference of genetic network architectures. In *Pacific symposium on biocomputing*, volume 3, pages 18–29, 1998.
13. Atul J Butte and Isaac S Kohane. Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements. In *Biocomputing 2000*, pages 418–429. World Scientific, 1999.
14. Alexander J Hartemink, David K Gifford, Tommi S Jaakkola, and Richard A Young. Using graphical models and genomic expression data to statistically validate models of genetic regulatory networks. In *Biocomputing 2001*, pages 422–433. World Scientific, 2000.

15. Timothy S Gardner, Diego Di Bernardo, David Lorenz, and James J Collins. Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, 301 (5629):102–105, 2003.
16. Adam A Margolin, Ilya Nemenman, Katia Basso, Chris Wiggins, Gustavo Stolovitzky, Riccardo Dalla Favera, and Andrea Califano. Aracne: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. In *BMC bioinformatics*, volume 7, page S7. Springer, 2006.
17. Mukesh Bansal, Giusy Della Gatta, and Diego Di Bernardo. Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. *Bioinformatics*, 22(7):815–822, 2006.
18. Curtis Huttenhower, K Tsheko Mutungu, Natasha Indik, Woongcheol Yang, Mark Schroeder, Joshua J Forman, Olga G Troyanskaya, and Hilary A Coller. Detailing regulatory networks through large scale data integration. *Bioinformatics*, 25(24):3267–3274, 2009.
19. Karen Lemmens, Tijl De Bie, Thomas Dhollander, Sigrid C De Keersmaecker, Inge M Thijs, Geert Schoofs, Ami De Weerdt, Bart De Moor, Jos Vanderleyden, Julio Collado-Vides, et al. Distiller: a data integration framework to reveal condition dependency of complex regulons in escherichia coli. *Genome biology*, 10(3):R27, 2009.
20. Aviv Madar, Alex Greenfield, Eric Vanden-Eijnden, and Richard Bonneau. Dream3: network inference using dynamic context likelihood of relatedness and the inferelator. *PloS one*, 5 (3):e9803, 2010.
21. Anne-Claire Haury, Fantine Mordelet, Paola Vera-Licona, and Jean-Philippe Vert. Tigress: trustful inference of gene regulation using stability selection. *BMC systems biology*, 6(1): 145, 2012.
22. Francesca Petralia, Pei Wang, Jialiang Yang, and Zhidong Tu. Integrative random forest for gene regulatory network inference. *Bioinformatics*, 31(12):i197–i205, 2015.
23. Nan Papili Gao, SM Minhaz Ud-Dean, Olivier Gandrillon, and Rudiyanto Gunawan. Sincerities: inferring gene regulatory networks from time-stamped single cell transcriptional expression profiles. *Bioinformatics*, 34(2):258–266, 2018.
24. Gourab Ghosh Roy, Nicholas Geard, Karin Verspoor, and Shan He. Polobag: Polynomial lasso bagging for signed gene regulatory network inference from expression data. *Bioinformatics*, 2020.
25. ENCODE Project Consortium et al. An integrated encyclopedia of dna elements in the human genome. *Nature*, 489(7414):57, 2012.
26. Robin Andersson, Claudia Gebhard, Irene Miguel-Escalada, Ilka Hoof, Jette Bornholdt, Mette Boyd, et al. An atlas of active enhancers across human cell types and tissues. *Nature*, 507(7493):455, 2014.
27. Magdalena Skipper, Alex Eccleston, Noah Gray, Therese Heemels, Nathalie Le Bot, Barbara Marte, and Ursula Weiss. Presenting the epigenome roadmap. *Nature*, 518:313, 2015.
28. Marine Louarn, Fabrice Chatonnet, Xavier Garnier, Thierry Fest, Anne Siegel, and Olivier Dameron. Increasing life science resources re-usability using semantic web technologies. In *Proceedings of the 15th IEEE International eScience conference, San Diego*, 2019.
29. T. Berners-Lee and J. Hendler. Publishing on the semantic web. *Nature*, 410(6832):1023–1024, 04 2001.
30. Tim Berners-Lee, Wendy Hall, James A. Hendler, Kieron O'Hara, Nigel Shadbolt, and Daniel J. Weitzner. A framework for web science. *Foundations and Trends in Web Science*, 1(1):1–130, 2007.
31. Christian Bizer, Tom Heath, and Tim Berners Lee. Linked data–the story so far. *International Journal on Semantic Web and Information Systems*, 5(3):1–22, 2009.
32. Maulik R Kamdar, Javier D Fernández, Axel Polleres, Tania Tudorache, and Mark A Musen. Enabling web-scale data integration in biomedicine through linked open data. *NPJ digital medicine*, 2:90, 2019. doi: https://doi.org/10.1038/s41746-019-0162-5.
33. Maulik R Kamdar and Mark A Musen. An empirical meta-analysis of the life sciences linked open data on the web. *Scientific data*, 8(1):24, 2021. doi: https://doi.org/10.1038/s41597-021-00797-y.
34. Aaron R Quinlan. Bedtools: the swiss-army tool for genome feature analysis. *Current protocols in bioinformatics*, pages 11–12, 2014.
35. Daniel Hernández, Aidan Hogan, and Markus Krötzsch. Reifying rdf: What works well with wikidata? *SSWS@ ISWC*, 1457:32–47, 2015.
36. Vinh Nguyen, Olivier Bodenreider, and Amit Sheth. Don't like rdf reification? making statements about statements using singleton property. In *Proceedings of the 23rd international conference on World wide web*, pages 759–770, 2014.
37. H. Yu, N. M. Luscombe, J. Qian, and M. Gerstein. Genomic analysis of gene expression relationships in transcriptional regulatory networks. *Trends Genet*, 19(8):422–427, Aug 2003.
38. Heonjong Han, Jae-Won Cho, Sangyoung Lee, Ayoung Yun, Hyojin Kim, Dasom Bae, Sunmo Yang, Chan Yeong Kim, Muyoung Lee, Eunbeen Kim, et al. Trrust v2: an expanded reference database of human and mouse transcriptional regulatory interactions. *Nucleic acids research*, 46(D1):D380–D386, 2018.
39. Luana Licata, Prisca Lo Surdo, Marta Iannuccelli, Alessandro Palma, Elisa Micarelli, Livia Perfetto, Daniele Peluso, Alberto Calderone, Luisa Castagnoli, and Gianni Cesareni. Signor 2.0, the signaling network open resource 2.0: 2019 update. *Nucleic acids research*, 48(D1): D504–D510, 2020.
40. Peter Langfelder and Steve Horvath. Wgcna: an r package for weighted correlation network analysis. *BMC bioinformatics*, 9(1):559, 2008.