Finding better partitions and conserved modules in Wnt signaling pathways

L. Nayak and R. K. De
Machine Intelligence Unit, Indian Statistical Institute, Kolkata, West Bengal, India

Abstract— *Human Wnt signaling pathway is involved in many crucial biological processes and its haywired behavior is found to be associated with various kinds of human cancers and other disorders. Modularized analysis will help in understanding modus operandi of this pathway. Here we partition the human Wnt signaling pathway into multiple partitions/modules by five algorithms inspired from different concepts. Greedy, Farhat's and Kernighan-Lin's algorithms are graph partitioning techniques. Newman's algorithm is dedicated towards finding communities in networks. Modularization algorithm detects functional modules in biological networks. A comparative study was done among partitions created by these algorithms by considering 'valid attribute' and 'functional enrichment' scores. Based on the functional enrichment score comparison, Modularization algorithm was found to create best partitions from the human Wnt signaling pathway. Later modules of 31 species-specific Wnt signaling pathways were studied and compared for detection of conserved modules.*

Keywords: KEGG, Gene Ontology, Modularization algorithm, Functional enrichment score

1. Introduction

The justification for dividing a network into a number of modules lies in the fact that the complexity of each module is much less than that of the entire pathway. It provides an easier means of studying the network by part. The task is difficult, because the components of a pathway always unite their mettle towards a common function. Hence, separating them into different classes/clusters/partitions or the latest term 'modules' is difficult. The partitions obtained as a result of the separation process is expected to upgrade existing knowledge and to simplify a task. There exist several methods for creating partitions from networks, but only a few of them have been applied to biological networks like graph partitioning algorithms and community finding algorithms. Methods based on graph partitioning algorithms ([1], [2], [3]) are rigid, as they demand cut number and cut-size information. It is not possible always to provide this information.

Community finding algorithms may help in finding existing communities in undirected metabolic pathways [4], directed networks [5] as well as overlapping community structure in DIP core list of protein-protein interactions of *S. cerevisiae* [6]. But, they have not been able to divide a network without existence of natural partition(s). Most of the biochemical networks come into this undividable category partially or fully. A newer flexible algorithm was required to overcome such kind of restrictions. The authors have devised an algorithm known as 'Modularization Algorithm' [7], in this regard. Modularization is a process by which one can split a network into smaller sub-networks called as modules. A module can be defined as a partition of the original network. It tends to be self-sufficient by maintaining minimal dependency on the rest part of the network. The algorithm is based on connectivity and topology of networks but does not require any cut-size or cut-number. It creates partitions from a network by using a complexity parameter 'c' [7]. It can also split a network without existence of any natural partition.

There are other existing partitioning approaches that can help towards designing an efficient modularization algorithm. A novel method to decompose biochemical networks is based on minimizing retroactivity among the created modules [8]. Retroactivity is the effect of the downstream elements on upstream elements. Another method claims to modularize biochemical networks based on classification of Petri net t-variants [9]. MOdularized NETwork learning (MONET) draws a whole network into overlapping modules and then tries to get the global picture by integrating the learned sub-networks [10]. Deterministic Modularization of Networks (dMoNet), a new agglomerative algorithm, finds even better modules in large-scale yeast and human protein interaction networks [11]. Bayesian networks and Probabilistic models are already used for identifying regulatory modules from gene expression data to identify functionally coherent modules and their correct regulators in *S. cerevisiae* [12]. Repeated random walk (RRW) based methods are used for discovering functional modules within large-scale protein interaction networks. They can find multi-functional proteins by allowing overlapping clusters [13].

Netsplitter [14] creates partitions progressively and the interactive visual matrix presentation allows considerable control over the process by the user, while incorporating special strategies to maintain the network integrity and minimize the information loss due to partitioning. Iterative Network Partition (iNP) identified modules in yeast protein complex network and breast cancer gene co-expression network [15]. Structural Clustering Algorithm for Networks

(SCAN) finds clusters or functional modules, hubs and outliers in complex biological networks [16]. Cartographic representation of networks can be used to find functional modules and uncover important new results in metabolic networks, such as the significant conservation of non-hub connector metabolites [17]. But none of them have been applied to signal transduction pathways. It will constitute an interesting work to combine these ideas to create a more robust partitioning algorithm and apply it to different kinds of pathways including that of signal transduction.

In this article, we have partitioned the human Wnt signaling pathway using various algorithms, *viz.*, Modularization algorithm [7], Newman's community finding algorithm [4], Greedy algorithm [1], Farhat's algorithm [2], and Kernighan-Lin's algorithm [3]. Their performances are compared based on 'valid attribute' and 'functional enrichment' scores in order to find the best partitioning algorithm. In addition, we have detected presence of conserved modules in 31 species-specific Wnt signaling pathways.

2. Materials and Methods

Here, we describe various partitioning algorithms. Then, we formulate a method for comparing these partitions by associating them to gene ontology terms. First of all, we describe different sources of pathway data.

2.1 Data

An exclusive list of all the signaling pathway databases is provided at http://www.pathguide.org/. Wnt signaling pathway data can be availed from some of these databases, *i.e.*, Reactome [18], BioCarta [19], PID [20], NetPath [21], STKE [22] and KEGG/PATHWAY [23] in various formats. No species-specific Wnt signaling pathway data is available other than hsa in PID and NetPath. Wnt data is available for hsa and mmu only in BioCarta. STKE has data for a few species (dme, dre, cel and hsa). In Reactome database Wnt signaling pathway information is available for 12 species. But, there is no option to download the molecular interactions of Wnt signaling pathway specific to each species. On the other hand, KEGG contains 48 species-specific Wnt signaling pathways (maximum number of species covered in any database at present). XML data files of the pathways along with their KGML and PNG diagrams are publicly accessible. We took 31 species-specific Wnt signaling pathways as raw data from this database (data taken in August 2009). These species-specific data are used for analysis in this work. Detailed information of these species is given in Table 1. The database uses a unique three letter code for each species along with their biological and common names (wherever applicable), *viz.*, 'hsa' for *H. sapiens* (human). These three letter codes are used extensively in this manuscript.

Table 1: Details of species taken from KEGG/PATHWAY database. For all these species, separate species-specific pathways are available in KEGG/PATHWAY database. The database uses a unique three letter code, *viz.*, 'hsa' for *H. sapiens* (human) for each species along with their biological and common names.

Sl. No.	Species Name	Common Name	KEGG code
01	*H. sapiens*	Human	hsa
02	*M. musculus*	Mouse	mmu
03	*R. norvegicus*	Rat	rno
04	*B. taurus*	Cow	bta
05	*C. familiaris*	Dog	cfa
06	*P. troglodytes*	Chimpanzee	ptr
07	*M. mulatta*	Rhesus Monkey	mcc
08	*M. domestica*	Opossum	mdo
09	*G. gallus*	Chicken	gga
10	*D. rerio*	Zebrafish	dre
11	*X. laevis*	African clawed frog	xla
12	*S. purpuratus*	Purple sea urchin	spu
13	*X. tropicalis*	Western clawed frog	xtr
14	*D. melanogaster*	Fruitfly	dme
15	*E. caballus*	Horse	ecb
16	*N. vectensis*	Sea anemone	nve
17	*A. mellifera*	Honey bee	ame
18	*D. pseudoobscura pseudoobscura*	-	dpo
19	*T. castaneum*	Red flour beetle	tca
20	*A. aegypti*	Yellow fever mosquito	aag
21	*O. anatinus*	Platypus	oaa
22	*C. elegans*	Nematode	cel
23	*A. gambiae*	Mosquito	aga
24	*S. scrofa*	Pig	ssc
25	*B. floridae*	Florida lancelet	bfo
26	*C. intestinalis*	Sea squirt	cin
27	*D. ananassae*	-	dan
28	*B. malayi*	Filaria	bmy
29	*A. pisum*	Pea aphid	api
30	*T. adhaerens*	-	tad
31	*C. briggsae*	-	cbr

2.2 Algorithms

We have used the Biological Networks Gene Ontology tool (BINGO) [24] for comparing performance among Modularization [7], Newman's community finding [4], Greedy [1], Farhat's [2], and Kernighan-Lin's [3] algorithms. C and Matlab (Version 7.0.4) have been used for implementation of these algorithms.

2.3 Scoring Method

BINGO is an open source java tool to determine the Gene Ontology (GO) terms that are significantly over-represented in a set of genes. GO [25] is a public consortium of databases that provides a controlled vocabulary of terms aiming at a gene's or a cluster of genes' biological annotations. It consists of three hierarchically structured sets of vocabularies that describe gene products in terms of their associated 'Biological Process (BP)', 'Molecular Function (MF)' and 'Cellular Component (CC)' information; 'Go Full (GF)'

being the superset of these sets. BINGO runs as a plug-in to Cytoscape [26]. BINGO retrieves the relevant GO annotations and propagates them upward through the GO hierarchy, *i.e.*, any gene annotated to a certain GO category is also explicitly included in all parental categories. It tries to answer the basic question, "While sampling X genes (test set) out of N genes (reference set), what is the probability that x or more of these genes belong to a functional category C shared by n of the N genes in the reference set?" Hypergeometric test answers this question in the form of a P-value. P-values depict a created partition's capability to lie in one category of biological function. If a particular partition created by a partitioning algorithm returns more number of valid GO terms with lower P-values than the others, the algorithm is believed as a better algorithm for creating partitions. Based on this belief, we have designed the 'valid attribute score'.

Valid attribute-wise analysis takes into consideration the number of valid GO attributes that the algorithm in consideration gets as result from a query with respect to a background database. Here, we have considered GO attributes obtained with P-value of the order of 10^{-5} or smaller as valid. The threshold P-value was fixed in such a manner that valid attributes from majority of the partitions can be collected. Counting the number of valid attributes that a partition is found to be associated with, is a well established way of determining the biological validity of that partition. Many clustering algorithms follow it as a comparative measure to establish their superiority among the others [27]. Here, we have considered three background databases, namely 'BP', 'CC' and 'GF'.

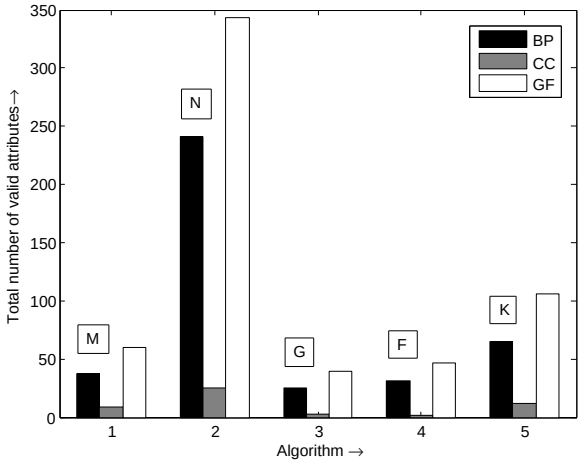
P-values give a good indication about the prominence of a certain functional category. But, no index of validity exists among the valid GO terms with lower P-values. Are they all equally valid or some of them are more valid than the others? Does such an index affect comparative results? By devising a validity index ('functional enrichment score'), the authors have showcased the change in results. Functional enrichment score-wise analysis takes into account functional enrichment scores of a set of partitioning algorithms. The functional enrichment score S_A of an algorithm A is defined as the mean of enrichment scores of the p partitions it has created.

$$S_A = \frac{1}{p}\sum_{i=1}^{p} S_{P_i} \tag{1}$$

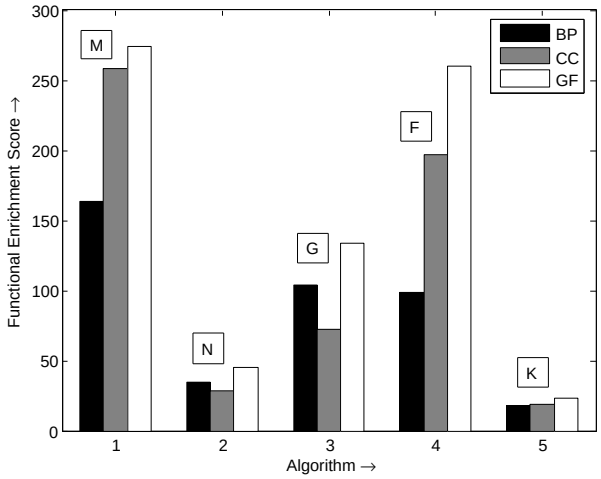
In turn the enrichment score S_{P_i} of a partition P_i is the average of the individual enrichment scores ($S_{T_{ij}}$) of associated individual attributes (T_{ij}s). Thus S_{P_i} is given by

$$S_{P_i} = \frac{1}{q}\sum_{j=1}^{q} S_{T_{ij}} \tag{2}$$

where q is the number of attributes. Enrichment score $S_{T_{ij}}$ of an individual attribute T_{ij} is calculated by comparing the performance of algorithm A with the performance of



(a)



(b)

Fig. 1: Performance comparison of various partitioning algorithms. (a) Comparison based on valid attribute score. Newman's community finding algorithm is performing better. (b) Comparison based on functional enrichment score of valid attributes. Modularization algorithm is performing better. [BP- Biological Process, CC- Cellular Component, GF- GO Full, M- Modularization algorithm, N- Newman's algorithm, G- Greedy algorithm, F- Farhat's algorithm and K- Kernighan-Lin's algorithm]

a background database in detecting over-expressed gene category(s) associated with the attribute. $S_{T_{ij}}$ depicts the efficiency of the partitioning algorithm in placing nodes having a common attribute in a partition with respect to a background database. Let x be the number of nodes associated with an attribute T_{ij}, which lies in a partition P_i, and X ($\geq x$) be the number of nodes present in partition P_i. Then x/X is the ability of an algorithm for placing nodes in a partition that are associated with attribute T_{ij}. Let y be the number of nodes associated with an attribute T_{ij} in a background database, and Y ($\geq y$) be the number of attributes in that database. Then y/Y is the ability of the background database to associate genes to attribute T_{ij}. Thus $S_{T_{ij}}$ can be defined as

$$S_{T_{ij}} = \frac{x}{X} / \frac{y}{Y} \tag{3}$$

In other words, we have taken the ratio of the performance of an algorithm with respect to the performance of a background database in assigning an attribute to a partition. Functional enrichment score is a measure to quantify the level of performance of an algorithm in creating biologically significant partitions. While comparing a few algorithms, higher the value of S_A, better is the algorithm for creating significant partitions. We have created three sets of enrichment scores, corresponding to three background databases, for each algorithm (Modularization, Newman's community finding, Greedy, Farhat's and Kernighan-Lin's) to get a better comparison.

3. Results and Discussions

Partitions of the human Wnt signaling pathway obtained by Modularization [7], Newman's community finding [4], Greedy [1], Farhat's [2], and Kernighan-Lin's [3] algorithms are described here. Human Wnt signaling pathway is a network of 60 nodes and 70 relations.

The best set of partitions created by each of the aforementioned algorithms was needed for the purpose of comparison. Hence, multiple sets of partitions were obtained by Modularization, Greedy and Farhat's algorithms where cut-number can be predesigned. Every individual set of partitions was evaluated by calculating their average functional enrichment score of associated valid attributes. The set of partitions having the highest functional enrichment score was deemed the best and used for comparison. The Modularization algorithm produced the best set of partitions (8 modules) for c = 3. Hence, one way of comparison was to create a set containing the same number of partitions from Greedy and Farhat's algorithm and then tally their average functional enrichment score of associated valid attributes. But, it would have been a biased way of comparison as some other set of partitions created by Greedy and Farhat's algorithm may yield a better functional enrichment score. So for Greedy and Farhat's algorithm, we have considered three immediate lower and higher cut-numbers including the cut-number 8 for creating sets of partitions [range: 5-11]. The best set among them (11 partitions for Farhat's algorithm and 9 partitions for Greedy algorithm) was used then for comparison. Newman's community finding algorithm created the best set of partitions (8 partitions) for ΔQ value of 1.0470e^{-017}. Two modules were generated by Kernighan-Lin's algorithm. The best sets of partitions created by all these algorithms are given in Table 2.

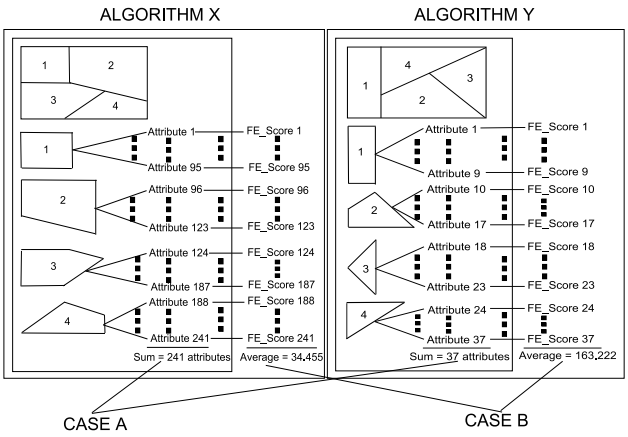


Fig. 2: **Methods of Algorithm Comparison.** (CASE A) Comparison based on valid attribute score shows that algorithm X is better than algorithm Y in creating partitions as the partitions are associated with more number of valid attributes. (CASE B) Comparison based on functional enrichment score of valid attributes shows that algorithm Y is better than algorithm X in creating partitions as the partitions are associated with some attributes, those have high association index (associated with more number of nodes in the partitions). Functional enrichment score is denoted as FE_Score. The later option is a better way in assigning biological significance to a partition, as the method of comparison can reflect the inner picture among the valid attributes rather than treating them as equals.

3.1 Performance Comparison of algorithms

The attribute-wise study takes into account the total number of valid attributes associated with the partitions obtained by an algorithm as a measure of their performance. Their overall performance is demonstrated in Figure 1(a). It shows that the Newman's community finding algorithm's partitions are returning maximum number of valid attributes (241, 25 and 343) with respect to all the three background databases namely 'BP', 'CC' and 'GF' followed by Kernighan-Lin's algorithm (65, 12 and 106). The next better performance is that of Modularization algorithm (37, 9 and 60) followed by Farhat's algorithm (31, 2 and 47) and Greedy algorithm (25, 3 and 39). Newman's algorithm is appeared to be the

Table 2: The best sets of partitions created by different partitioning algorithms. All the results are based on human Wnt signaling pathway data. Entries list the nodes in a partition. [Farhat's algorithm: 11 partitions; Greedy algorithm: 9 partitions; Modularization algorithm: c = 3, 8 modules; Newman's community finding algorithm: 8 partitions, ΔQ = 1.0470e^{-017}; Kernighan-Lin's algorithm: 2 partitions, initial cut-size 8, final cut-size 4]

Partition No.	Farhat	Greedy	Modularization	Newman	Kernighan-Lin
01	PSEN1, CTNNB1, PRKACA, CTNNBIP1, CHD8, SIAH1	PSEN1, CTNNB1, PRKACA, GSK3B, AXIN1, CSNK1A1L, SIAH1, TP53	LEF1, SMAD4, NLK, SOX17, CTBP1, CREBBP, RUVBL1, MYC, JUN, FOSL1, CCND1, PPARD, MMP7, MAP3K7	MAPK8, RAC1, ROCK1, RHOA, DAAM1, DVL1, PRICKLE1, FZD10, VANGL2, WNT9A	DKK1, PORCN, LRP6, CER1, WIF1, FZD10, WNT16, FZD10, SFRP1, WNT9A, VANGL2, PRICKLE1, WNT5A, PRKCA, DVL1, RAC1, DAAM1, FZD10, MAPK8, RHOA, ROCK1, PLCB1, CHP, CAMK2A, NFAT5, SIAH1, TP53, NKD1, JUN, FOSL1
02	GSK3B, DVL1, AXIN1, FRAT1, FZD10	APC2, DVL1, TBL1X, FRAT1, FZD10, CXXC4	CTNNB1, PSEN1, CTNNBIP1, CHD8, PRKACA, CSNK1A1L, FBXW11, TBL1X	WIF1, CER1, PORCN, WNT16	CCND1, MMP7, MYC, PRKACA, PSEN1, TBL1X, APC2, CTNNB1, GSK3B, AXIN1, FBXW11, DVL1, CSNK1E, PPP2CA, CTNNBIP1, CHD8, FRAT1, CXXC4, SENP2, CSNK2A1, CSNK1A1L, RUVBL1, MAP3K7, PPARD, NLK, SMAD4, LEF1, SOX17, CTBP1, CREBBP
03	WNT16, SFRP1, LRP6, PORCN, CER1	SENP2, NKD1, DVL1, FZD10, RAC1, DAAM1	DVL1, CXXC4, SENP2, CSNK2A1, FRAT1, APC2, NKD1	DKK1	–
04	DKK1, PPARD, APC2, SMAD4, LEF1	VANGL2, PRICKLE1, WNT9A, FBXW11, CSNK2A1, PPP2CA	WNT16, PORCN, FZD10, SFRP1, CER1, WIF1, LRP6, DKK1	SIAH1, TP53	–
05	NLK, SOX17, CTBP1, CREBBP, RUVBL1	CSNK1E, RUVBL1, LEF1, SMAD4, NLK, SOX17	DVL1, FZD10, RAC1, DAAM1, VANGL2, PRICKLE1, WNT9A, MAPK8, RHOA, ROCK1	NFAT5, PRKCA, CHP, CAMK2A, PLCB1, FZD10, WNT5A	–
06	MAP3K7, MMP7, WIF1, CXXC4, SENP2	CTBP1, MMP7, PPARD, CCND1, FOSL1, JUN	AXIN1, CSNK1E, GSK3B, PPP2CA	MMP7, PPARD, CCND1, FOSL1, JUN, MYC, CTBP1, SOX17, SMAD4, CREBBP, RUVBL1, NLK, MAP3K7, LEF1	–
07	APC2, TBL1X, CSNK1A1L, CCND1, FOSL1	MYC, CREBBP, CHD8, CTNNBIP1, LRP6, DKK1	PLCB1, FZD10, CAMK2A, CHP, PRKCA, WNT5A, NFAT5	CHD8, CTNNBIP1, CSNK1A1L, FBXW11, TBL1X, AXIN1, PPP2CA, APC2, CTNNB1, PRKACA, PSEN1, GSK3B, CSNK1E	–
08	JUN, MYC, NKD1, DVL1, FZD10	SFRP1, WNT16, PORCN, CER1, WIF1, MAPK8	TP53, SIAH1	NKD1, FRAT1, SENP2, DVL1, CSNK2A1, CXXC4, LRP6, FZD10, SFRP1	–
09	WNT9A, PRICKLE1, VANGL2, DAAM1, RHOA	RHOA, ROCK1, MAP3K7, WNT5A, FZD10, PLCB1, PRKCA, CHP, NFAT5, CAMK2A	–	–	–
10	MAPK8, ROCK1, RAC1, FBXW11, CSNK2A1	–	–	–	–
11	PPP2CA, CSNK1E, WNT5A, FZD10, PLCB1, PRKCA, CHP, NFAT5, CAMK2A	–	–	–	–

best algorithm for creating partitions as they are found to be associated with the highest number of valid attributes.

At a deeper level we have found that small subsets of a large partition were always found to be associated with many attributes (Figure 2). A large partition ensures presence of many subsets in it, which are associated with GO attributes; some of them being unique. Thus the corresponding P-values will be lower and they will be considered as valid. But only validity of an attribute is not sufficient for defining goodness of a module. Ideally, a valid attribute must be given more preference if it is associated with more number of nodes present in a partition than another one associated with less number of nodes in the same partition. In other words, we needed to know the number of attributes that actually show some goodness (associated with more number of nodes) in justifying a partition. Hence, a functional enrichment score system was defined to give weightage to valid attributes according to their goodness of performance.

Functional enrichment score depicts the efficiency of a partitioning algorithm in placing nodes (having a common attribute) in a partition with respect to a background database. Higher the value of the score, better is the algorithm for creating partitions. The average enrichment scores (S_As) (Equation 1) of the different algorithms are shown in Figure 1(b). The Modularization algorithm is found to be performing best among all the algorithms considered here, courtesy this figure. The algorithm creates partitions with average functional enrichment score of 163.22, 258.37, and 274.19 with respect to 'BP', 'CC' and 'GF' as background databases. Kernighan-Lin's algorithm creates partitions with the least average functional enrichment score (17.66, 18.93 and 23.14 respectively) preceded by Newman's algorithm (34.45, 28.08 and 44.79 respectively), although, both the algorithms have created partitions for which maximum number of valid attributes are found to be associated (Figure 1(a)). It proves that only counting valid attributes associated with a partition is not a proper measure to deem that partition as good. Among the valid attributes, an association index must be established. Functional enrichment scores reflect such association index. Among Greedy and Farhat's algorithms,

Table 3: Module information of species-specific Wnt signaling pathways. Details about the individual Wnt signaling pathway modules of different species are given in this table [n: number of connected nodes in a species-specific pathway; r: number of relations present the connected component of a species-specific pathway; t: total number of modules created from a species-specific pathway]. The modules have been created for c = 3. The table throws light on the developmental trend of Wnt signaling pathway among the taken set of species. Number nodes present in each module is listed along side it in parentheses.

c	n	r	t	WNT	(DVL)1	Axin	β-catenin	TCF	Tp53	(DVL)2	PLC
hsa	60	70	8	WNT [8]	(DVL)1 [7]	Axin [4]	β-catenin [8]	TCF [14]	p53 [2]	(DVL)2 [10]	PLC [7]
mmu	60	70	8	WNT [8]	(DVL)1 [7]	Axin [4]	β-catenin [8]	TCF [14]	p53 [2]	(DVL)2 [10]	PLC [7]
rno	59	69	8	WNT [7]	(DVL)1 [7]	Axin [4]	β-catenin [8]	TCF [14]	p53 [2]	(DVL)2 [10]	PLC [7]
bta	58	68	8	WNT [7]	(DVL)1 [6]	Axin [4]	β-catenin [8]	TCF [14]	P53 [2]	(DVL)2 [10]	PLC [7]
cfa	58	68	8	WNT [8]	(DVL)1 [7]	Axin [4]	β-catenin [7]	TCF [13]	p53 [2]	(DVL)2 [10]	PLC [7]
ptr	58	67	8	WNT [8]	(DVL)1 [7]	Axin [4]	β-catenin [8]	TCF [13]	p53 [2]	(DVL)2[10]	PLC [6]
mcc	55	63	8	WNT [7]	(DVL)1 [6]	Axin [4]	β-catenin [8]	TCF [13]	p53 [2]	(DVL)2 [8]	PLC [7]
mdo	54	64	7	WNT [8]	(DVL)1 [7]	Axin [2]	β-catenin [9]	TCF [11]	-	(DVL)2 [10]	PLC [7]
gga	54	63	8	WNT [7]	(DVL)1 [6]	Axin [3]	β-catenin [8]	TCF [11]	p53 [2]	(DVL)2 [10]	PLC [7]
dre	52	60	7	WNT [8]	-	Axin [4]	β-catenin [7]	TCF [13]	p53 [2]	(DVL)2 [11]	PLC [7]
xla	43	45	6	WNT [7]	-	-	β-catenin [8]	TCF [11]	p53 [2]	(DVL)2 [8]	PLC [7]
spu	39	45	6	-	(DVL)1 [7]	Axin [2]	β-catenin [5]	TCF [10]	-	(DVL)2 [9]	PLC [6]
xtr	37	36	6	WNT [3]	-	-	β-catenin [7]	TCF [6]	p53 [2]	(DVL)2 [12]	PLC [7]
dme	36	42	7	WNT [6]	(DVL)1 [5]	Axin [2]	β-catenin [6]	TAK1 [2]	-	(DVL)2 [9]	PLC [6]
ecb	36	38	7	(Frizzled)1 [5]	(DVL)1 [3]	-	β-catenin [9]	TAK1 [2]	p53 [2]	(DVL)2 [8]	PLC [7]
nve	32	33	6	(Frizzled)1 [5]	-	Axin [5]	β-catenin [6]	TAK1 [2]	-	(DVL)2 [7]	PLC [7]
ame	30	32	5	-	(DVL)1 [4]	-	β-catenin [9]	TAK1 [2]	-	(DVL)2 [8]	PLC [7]
dpo	28	30	4	(Frizzled)1 [8]	-	-	β-catenin [7]	-	-	(DVL)2 [8]	PLC [5]
tca	26	27	4	(Frizzled)1 [7]	-	-	β-catenin [7]	-	-	(DVL)2 [6]	PLC [6]
aag	24	22	4	(Frizzled)1 [4]	-	-	β-catenin [4]	-	-	(DVL)2 [10]	PLC [6]
oaa	22	22	4	WNT [2]	-	-	β-catenin [7]	-	-	(DVL)2 [8]	PLC [5]
cel	22	20	3	-	-	-	β-catenin [10]	-	-	RhoA [6]	PLC [6]
aga	20	18	3	(Frizzled)1 [11]	-	-	β-catenin [4]	-	-	-	PLC [5]
ssc	19	16	4	FRP [2]	-	-	β-catenin [5]	TCF [7]	-	-	PLC [5]
bfo	18	16	3	-	-	-	β-catenin [9]	-	-	(DVL)2 [5]	PLC [4]
cin	17	14	3	-	(DVL)1 [7]	-	-	-	-	(DVL)2 [5]	PLC [5]
dan	16	12	4	-	(DVL)1 [2]	-	β-catenin [4]	-	-	(DVL)2 [5]	PLC [5]
bmy	13	11	3	-	(DVL)1 [4]	-	-	-	-	(DVL)2 [5]	PLC [4]
api	13	10	3	-	(DVL)1 [4]	-	-	-	-	(DVL)2 [5]	PLC [4]
tad	6	4	2	-	-	-	-	-	-	Rac [2]	PLC [4]
cbr	4	3	1	-	(DVL)1 [4]	-	-	-	-	-	-

the later performs better for the background databases 'BP' (103.65), while Greedy algorithm creates partitions that are found to be associated with more attributes of 'CC' and 'GF' databases (197.35 and 260.06).

3.2 Comparative analysis to find conserved modules

Here, modules of 31 species-specific Wnt signaling pathways (aag, aga, ame, api, bfo, bmy, bta, cbr, cel, cin, cfa, dan, dmw, dpo, dre, ecb, gga, hsa, mcc, mdo, mmu, nve, oaa, ptr, rno, ssc, spu, tad, tca, xla and xtr) were analyzed and subjected for comparative analysis. Module details of these 31 species are given in Table 3. It is important to mention here that the Wnt signaling pathway of each species may vary in terms of nodes, relations and topology. More number of absent nodes depict a pathway's lower level of development. Likewise more number of isolated nodes indicate towards poorly developed architecture of a pathway. But, in some cases nodes or relations *absentia* do mean lack of information to indicate their presence in a pathway. Table 3 gives individual details (number of nodes, relations and modules) of all the pathways considered here.

Wnt signaling pathways of the aforementioned species were subjected to modularization for c = 3 as for the same c-value meaningful modules were found in human Wnt

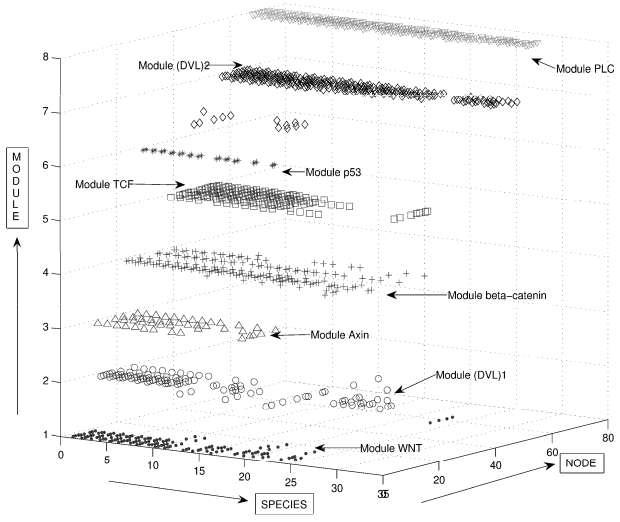


Fig. 3: One to one module wise comparison of Wnt signaling pathway of 31 different species

signaling pathway. We were getting 2 to 8 modules for each species that vary in their size (number of nodes present in the module) as shown in Table 3. Modules Wnt and $\beta-catenin$ were found to be conserved in 9 species (hsa, mmu, rno, bta, cfa, ptr, mcc, mdo and gga), module TCF was found to be conserved in 5 species (hsa, mmu, rno, bta and cfa); module $Tp53$ was observed in altogether 12 species (hsa, mmu, rno, bta, cfa, ptr, mcc, gga, dre, xla, xtr and ecb) and it was conserved by size and topology in all these species; module $(DVL)2$ remained conserved in 11 species (hsa, mmu, rno, bta, cfa, ptr, mdo, gga, dre, spu and dme); module PLC turned out to be the most conserved module, found in a maximum number of 17 species (hsa, mmu, rno, bta, cfa, ptr, mcc, mdo, gga, dre, xla, spu, xtr, dme, ecb, nve and ame). Conservation patterns are shown in Figure 3.

4. Conclusions

Modularization algorithm is a better algorithm to create modules from human Wnt signaling pathway. A new GO attribute based score (Functional enrichment score) is designed for validating these modules. The score establishes a validity index among GO attributes and can be extended for performance measurement of any kind of partitions/clusters/modules created from biological networks. A comparative study of 31 species-specific Wnt signaling pathway modules is done by utilizing this algorithm. Module PLC is found to be the most conserved module, found in a maximum number of 17 species. Wnt signaling pathway is found to be intrinsic in many diseases; being a major player in the human cancer arena. Hence, knowledge about conserved modules can be utilized in laboratory experiments when a particular module is found to be associated with the background mechanism of a disease.

References

[1] G. Chartrand and O. R. Oellermann, *Applied and algorithmic graph theory*. New York: McGraw Hill, 1993.

[2] C. Farhat, "A simple and effcient automatic FEM domain decomposer," *Computers and Structures*, vol. 28, pp. 579–602, 1988.

[3] B. W. Kernighan and S. Lin, "An Efficient Heuristic Pprocedure for Partitioning Graphs," *The Bell System Technical Journal*, vol. 49, pp. 291–307, 1970.

[4] M. E. J. Newman, "Modularity and community structure in networks," in *Proc Natl Acad Sci.* PNAS, USA, 2006, pp. 8577–8482.

[5] E. A. Leicht and M. E. J. Newman, "Community Structure in Directed Networks," *Physical Review Letters*, vol. 100, no. 118703, 2008.

[6] G. Palla, I. Derenyi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, pp. 814–818, 2005.

[7] L. Nayak and R. K. De, "An algorithm for modularization of MAPK and calcium signaling pathways: Comparative analysis among different species," *Journal of Biomedical Informatics*, vol. 40, pp. 726–749, 2007.

[8] J. Saez-Rodriguez, S. Gayer, M. Ginkel, and E. D. Gilles, "Automatic decomposition of kinetic models of signaling networks minimizing the retroactivity among modules," *Bioinformatics*, vol. 24, pp. i213–i219, 2008.

[9] E. Grafahrend-Belau, F. Schreiber, M. Heiner, A. Sackmann, B. H. Junker, S. Grunwald, A. Speer, K. Winder, and I. Koch, "Modularization of biochemical networks based on classification of Petri net t-invariants," *BMC Bioinformatics*, vol. 9, no. 90, 2008.

[10] P. H. Lee and D. Lee, "Modularized learning of genetic interaction networks from biological annotations and mRNA expression data," *Bioinformatics*, vol. 21, pp. 2739–2747, 2005.

[11] R. L. Chang, F. Luo, S. Johnson, and R. H. Scheuermann, "Deterministic GraphŰTheoretic Algorithm for Detecting Modules in Biological Interaction Networks," *International Journal of Bioinformatics Research and Application*, vol. 6, no. 6, pp. 101–119, 2010.

[12] E. Segal, M. Shapira, A. Regev, D. Pe'er, D. Botstein, D. Koller, and N. Friedman, "Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data," *Nature Genetics*, vol. 34, pp. 166–176, 2003.

[13] K. Macropol, T. Can, and A. K. Singh, "RRW: repeated random walks on genome-scale protein networks for local cluster discovery," *BMC Bioinformatics*, vol. 10, p. 283, 2009.

[14] W. S. Verwoerd, "A new computational method to split large biochemical networks into coherent subnets," *BMC Systems Biology*, vol. 5, no. 25, 2011.

[15] S. Sun, X. Dong, Y. Fu, and W. Tian, "An iterative network partition algorithm for accurate identification of dense network modules," *Nucleic Acids Research*, vol. 40, no. 3, p. e18, 2012.

[16] M. Mete, F. Tang, X. Xu, and N. Yuruk, "A structural approach for finding functional modules from large biological networks," *BMC Bioinformatics*, vol. 9(Suppl 9), no. S19, 2008.

[17] R. Guimera and L. A. N. Amaral, "Functional cartography of complex metabolic networks," *Nature*, vol. 433, pp. 895–900, 2005.

[18] G. Joshi-Tope, M. Gillespie, I. Vastrik, P. DEustachio, E. Schmidt, B. D. Bono, B. Jassal, G. R. Gopinath, G. R. Wu, L. Matthews, S. Lewis, E. Birney, and L. Stein, "Reactome: a knowledgebase of biological pathways," *Nucleic Acids Research*, vol. 33, pp. D428–D432, 2005.

[19] D. Nishimura, "A view from the Web: Biocarta," *Biotech. Software and Internet Report*, vol. 2, no. 3, pp. 117–120, 2001.

[20] C. F. Schaefer, K. Anthony, K. S, J. Buchoff, M. Day, H. T, and B. K. H, "PID: The Pathway Interaction Database," *Nucleic Acids Res.*, vol. 37, pp. D674–D679, 2009.

[21] K. Kandasamy, S. S. Mohan, R. Raju, S. Keerthikumar, G. S. S. Kumar, A. K. Venugopal, D. Telikicherla, J. D. Navarro, S. Mathivanan, C. Pecquet, S. K. Gollapudi, S. G. Tattikota, S. Mohan, H. Padhukasahasram, Y. Subbannayya, R. Goel, H. K. C. Jacob, J. Zhong, R. Sekhar, V. Nanjappa, L. Balakrishnan, R. Subbaiah, Y. L. Ramachandra, B. A. Rahiman, T. S. K. Prasad, J. Lin, J. C. D. Houtman, S. Desiderio, J. Renauld, S. N. Constantinescu, O. Ohara, T. Hirano, M. Kubo, S. Singh, P. Khatri, S. Draghici, G. D. Bader, C. Sander, W. J. Leonard, and A. Pandey, "NetPath: a public resource of curated signal transduction pathways," *Genome Biology*, vol. 11, p. R3, 2010.

[22] K. I. Fukuda and T. Takagi, "Knowledge Representation of Signal Transduction Pathways," *Bioinformatics*, vol. 17, pp. 8290–837, 2001.

[23] M. Kanehisa and S. Goto, "KEGG: Kyoto Encyclopedia of Genes and Genomes," *Nucleic Acids Research*, vol. 28, pp. 27–30, 2000.

[24] S. Maere, K. Heymans, and M. Kuiper, "BiNGO: a Cytoscape plugin to assess overrepresentation of Gene Ontology categories in Biological Networks," *Bioinformatics*, vol. 21, pp. 3448–3449, 2005.

[25] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock, "Gene Ontology: tool for the unification of biology," *Nature Genetics*, vol. 25, pp. 25–29, 2000.

[26] P. Shannon, A. Markiel, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker, "Cytoscape: A Software Environment for Integrated Models of Biomolecular Interaction Networks," *Genome Res.*, vol. 13, pp. 2498–2504, 2003.

[27] A. Bhattacharya and R. K. De, "Divisive Correlation Clustering Algorithm (DCCA) for grouping of genes: detecting varying patterns in expression profiles," *Bioinformatics*, vol. 24, pp. 1359–1366, 2008.