### Using Treemaps to Visualize Gene Ontologies Ketan Babaria Human Computer Interaction Lab and Institute for Systems Research University of Maryland, College Park, MD USA 12/04/2001

### 1. Introduction

Treemaps are a space-filling visualization for hierarchical structures that show attributes of leaf nodes by size and color-coding. Treemaps enable users to compare sizes of nodes and of sub-trees, and are especially helpful in revealing patterns.

The Gene Ontology Consortium maintains Gene Ontology for molecular functions, biological processes and cellular components of gene products. These ontologies can be represented as a hierarchy.

This paper describes how using the gene ontology hierarchy coupled with the treemap visualization technique can be used to effectively visualize genome data.

### 2. Gene Ontology

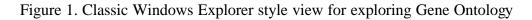
#### 2.1 Objective

The objective of Gene Ontology (GO) is to provide controlled vocabularies for the description of the molecular function, biological process and cellular component of gene products. The controlled vocabularies of terms are structured to allow both attribution and querying to be at different levels of granularity. The three organizing principles of GO are molecular function, biological process and cellular component [1]. Each of these are further divided into subparts. For example cellular component is further divided into cell, external protective structure, extracellular, obsolete, unlocalized etc. Each of these nodes is further divided into sub categories.

#### 2.2 Visualizing GO

For visualizing the GO hierarchy, the web site uses Windows Explorer style information seeking (Figure 1). The figures on the right show number of genes of a particular type (e.g. fly genes) in each hierarchy. If the users want to see the nodes inside cellular component, they have to click on cellular component and the web page will show the nodes inside it (Figure 2.)

Hierarchy for <b>Gene_Ontology</b> and sub-parts Component of:			
Term	Fly Genes	Mouse Genes	Sacc. Yeast Genes
=== <u>v</u> Gene Ontology	7013	<u>6111</u>	<u>6399</u>
<u>v</u> cellular component	3121	4005	2439
v molecular function	<u>6345</u>	5003	<u>5813</u>
<u>v</u> <u>biological_process</u>	<u>2774</u>	<u>3897</u>	<u>5716</u>
	styles:v,> have su f,  is an instance of, ~~	· · ·	



Term	Fly Genes	Mouse Genes	Sacc. Yeast Genes
=== <u>v</u> <u>Gene_Ontology</u>	7013	<u>6111</u>	<u>6399</u>
<u>v</u> <u>cellular_component</u>	3121	<u>4005</u>	2439
<u>&gt;</u> <u>cel1</u>	<u>2933</u>	<u>3683</u>	<u>1950</u>
<u>cellular_component_unknown</u>	0	<u>178</u>	<u>436</u>
> external protective structure	0	0	<u>40</u>
<u>&gt;</u> extracellular	<u>133</u>	<u>140</u>	24
<u>&gt;</u> <u>obsolete</u>	_8	0	0
$\dots \ge unlocalized$	<u>_8</u> 90	22	<u>17</u>
<u>v</u> molecular_function	<u>6345</u>	<u>5003</u>	<u>5813</u>
$\dots \geq anti-toxin$	0	0	0
<u>anticoagulant</u>	0	_2	0
<u>&gt;</u> antifreeze	0	0	0
<u>&gt;</u> <u>antioxidant</u>	<u>_8</u>	_2	_2
antisense RNA	0	0	0
<u>&gt;</u> apoptosis regulator	<u>14</u>	_5	0
> cell adhesion molecule	<u>53</u>	<u>45</u>	_3
> <u>cell cycle regulator</u>	<u>17</u>	<u>128</u>	<u>3</u> <u>13</u>
> chaperone	53 <u>17</u> <u>114</u>	<u>5</u> <u>45</u> <u>128</u> <u>57</u>	<u>60</u>
<u>&gt;</u> <u>chaperone regulator</u>	0	0	0
cytoskeletal regulator	_1	<u></u>	0
> defense/immunity protein	46	256	_1

Figure 2. Figure showing the 2<sup>nd</sup> level hierarchy for GO

Such representation provides detailed content information and it's especially helpful when the users want to scan through all of the listings. However the use of such a representation makes it difficult to extract information as the navigation of the structure is a burden and the content information is often hidden within individual nodes. The problem becomes more acute when there is more than one attribute for leaf nodes.

## 3. Treemaps

Treemaps graphically represent hierarchical information via a two-dimensional rectangular map, providing compact visual representations of complex data spaces through both area and color [2]. The treemap can represent both hierarchical structure and each element's quantitative information simultaneously utilizing 100% of the designated screen area. Application arenas for treemaps have included computer directory browsing, stock market portfolio visualizations, an NBA player statistical browser and a budget viewer. For example, Figure 3 shows a treemap for visualizing stock data [3]. The size of boxes show the market capitalization and the color show whether the stock prices went up (green) or down (red). We see that the market capitalization of Exxon Mobile is highest in Energy sector and on this day the stock went down.

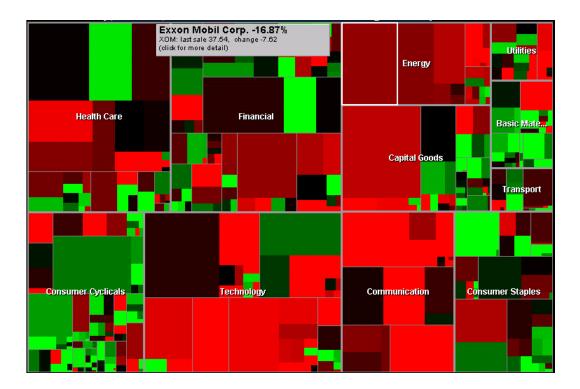


Figure 3. Treemap showing the performance of stock market.

# 4. Treemap representation of GO

Since the Gene Ontology can be represented by hierarchical trees, these trees translate directly to the treemap visualization method. Figure 4 is an example of a treemap generated with Treemap 3.0 software for the same GO hierarchy that was shown in Figure 3. A base rectangle represents the root of the GO hierarchy and is divided into small rectangular areas proportional to their relative importance. Users can size by any numerical attribute of the leaf node. The leaf nodes in this case are the genes. So the user can size by one of its attributes such as average fold change.

GO														
molecular	Cellular_Component													
antioxidant	antisense_RNA	apoptosis_regulat	Cell	Cellular_Compon										
cell_adhesion_mo	chaperone	chaperone_regulat	external+protectiv	extracellular										
cell_cycle_regulat	cytoske;etal_regul	defense_immunity	obsolete	unlocalised										
c	ntioxidant :ell_adhesion_mo	molecular_function intioxidant antisense_RNA :ell_adhesion_mo chaperone	molecular_function   untioxidant antisense_RNA apoptosis_regulat   sell_adhesion_mo chaperone chaperone_regulat	molecular_function Cellular_C   untioxidant antisense_RNA apoptosis_regulat Cell   :ell_adhesion_mo chaperone chaperone_regulat external+protectiv										

While the Windows Explorer representation of GO data is useful, quantitative attributes of genes may be better displayed and queried using treemaps. For example if the genome is analyzed for genes that are transcribed during cell death, 3087 genes are expressed at different levels. The changes in gene level (fold change) and the representation of this information, and how genes are represented in the larger context of GO demonstrate the utility of treemap for this application.

In this dataset each of 3087 genes has a number of attributes (average fold change, base pairs, pH value, etc). For approximately half of the genes in the dataset, their position in the GO hierarchy is known and the rest are labeled as unknown. The user may want to know the distribution of this data set in the GO hierarchy. Figure 5 shows the treemap for this data. Each square represents a gene. Using the treemap a user can recognize that almost 50% of the genes fall in the unknown category and approximately 25% of the genes fall under the enzymes category and so on.

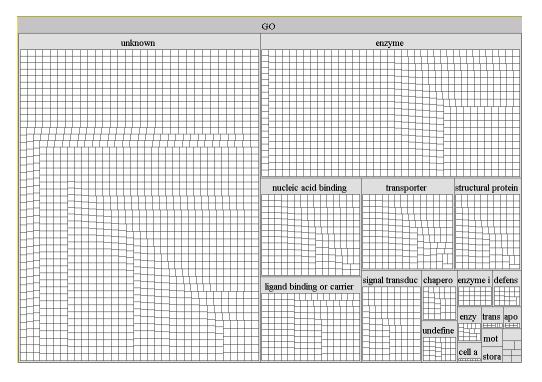


Figure 5 Gene distributions in each GO category using Treemap

Now if the user has information regarding the second level of the hierarchy for each gene, the treemap becomes more effective in visualizing the distribution. Figure 6 shows the distribution of genes in two levels of hierarchy. Such distribution can be extended to any level of the hierarchy.

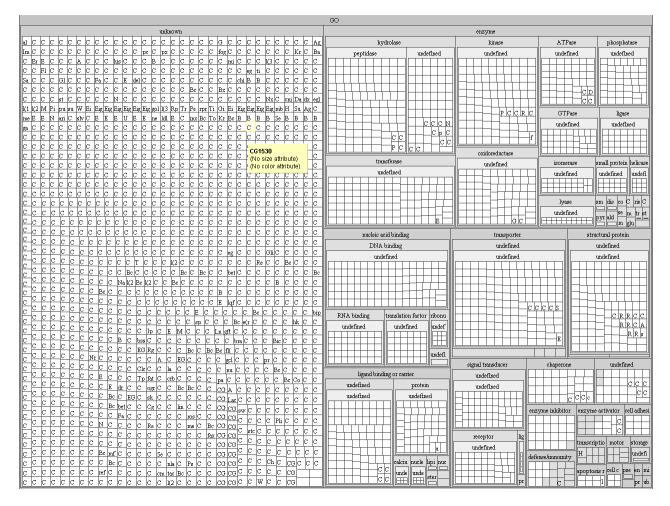


Figure 6 Gene distributions in 2<sup>nd</sup> level GO category using Treemap

In this example hydrolase has highest number of genes in enzyme group.

## 5. Data manipulation tools

## 5.1 Sizes and Color

In the GO hierarchy all of the leaf nodes (genes) have the same number of attributes. The attributes can be numerical (for example average fold change) and non numerical (for example acidic or basic) users can then modify the color and sizes of the nodes depending on a particular criterion. In the above example users can specify the size by criteria to be 'Average Fold Change'. Figure 7 shows the resulting treemap.

														GO														4		Attribute			alue
						unk	nown												_	enzyme								٦	(Nar	me)		CG485	)
	06											aa						1			-				c   a	~ ~ ~ ~			(Size			217.34	445190.
CG155	05			ľ	CG12	643		- IC	G24	44		CG4	1091		CG4859	'	CG7402	puc		CG365	υk	2G78L	16  C	G176	6  C	G102	CG14	0000	GoH	lierarchy			
																			_								L	0000	Aver	ageFoldC	h	2	17.3444
																		CG16	C	G10 CG	13	CG13	PR.	La	lpha	Pez	CG7	2000	Incre	easeDecr	e		
																	CG5919		┢		_	L	4	+		1		- 2000					
										_								RhoB	C	G8 mnb	C	39 Bc	DD	de N	Ic (	cg  c	G alp	0000					
CG722	4		CC	8788		c	CG153	309		CG	1600		CG	7447	CG5191	. 1	CG3132	1	ta	rc4 CG	Bc	am	cgl	c k	c İd	: sm	Ic C	2000	A 7				
																ſ	000102	CG76	5			; C					c c		Mai	in Leger		iltore	
																ļ			-6	G1	μ				Υ	Ϋ́́Υ	ΥĽ		IAISU	Leger		Inters	
																	Doa	Mlc-k		PR2	cc	; -	t	ĹĽ	Ĺ	11	11		Lab	el: (Nan	ne)		
CG181	35	CG120	012	CG38	338	CG1	021	BG	DS0		<b>3</b> 186	CG1	46	CG123	CG5171	.		<u> </u>	F	dh <sub>slm</sub>	bet	H	_		┼┼	+++	+++	- 8					
									-							ł		Rac2		G9 CG	P.	╈	┢	HÞ	Ħ	111			Size	e: Avera	geFc	ldChan	je
																	EG_25E8		41				+	HÞ	Ħ	111	<u>, tt † †</u>			_			
CG140	66 C	G7500	C	31035	ca	2297	CGé	610	CG1	on l	brat		130	CG162	11	_		BCDN	c	G1 CG	β	吘	Ē	H	±F		ÉÉÉÉ	E	Col	o <mark>r:</mark> (Non	e) -		
00140		07570	ľ	1055	00	4477	1000		p	1	orai	<b>P</b>	1.57	00102	CG5505		nmo	RnrS	Ŀ	CG	cc	١T	┺	P	±Ρ	18		HB	-NI	umerical	Colo	rina —	
			4				-	_	L		_		_	L				land	P	G1 Ubc	Ał	υŪ	┢	HF	FF	##		副					
CG11	EG_	2  CG:	59 C	G15	CG5	CG1	CG:	1 da	on H	BcD	CG1	CG1	CC.	19 CG8		liman	d binding	or comi			T		nuck	vic ac	rid b	inding		- 8		Negative		Zero	Pos
		1		1	1							1	1	-			in onioniß		_		٦lr			-		numig		18					
CG73	CG1	CG1	CHS	CG1	CGS	s cg:	2 CG	CC	;  cc	3  C	G C	3  CC	3  C	G fog	CG1168	S		CG102	23	/		CG3	bun	So	H								
	CG6	EG e	egh (	ce le	G C	3 CG	; CG	CG	CG	CG	CG	cg d	G	CG CG									CG3	+	다		++++	- 8					
CG 59	200	CG						- T	c	C.		_	T	c c	1							CG1	CG2	С	H								
0000	CG4	1 1						<b>T</b> <sup>1</sup>	ari C				1 <u>-</u>	wc	11								CGS	c	口	1115	目曲	<u></u>					
				<u>sec</u>	17 1			Ϋ́	$\frac{an}{1}$	; [			E	TW IC	1						H			1.			1-1114	≝ 8					
CG17	CG9	CG		3i B	+	-1-1	┢╼┢╸	τt			44		гŤ		bif	CG3	57 CG1	nuf Ci	ra I	CG C77	٦,			-	_	lucer							
	CG1	1		7 <u>G</u>	++			Ηt	11		τ'n	т'n	Ψ	++	10.	003		cnn T	Ī		-116	CG41	4 (C)	G3 🛛	CG		+++						
CG93	COL	Cor	CG	<u>'G</u>	-			Ţ	ĽГ	ŢΊ		ΤÌ	Ì.		1	-				- PhC	╢		$\vdash$		CG	_++	+++						
	CG6		cg	xg⊢					$\left  \right $		$\square$	++	++	+++		CG1	.06 CG5	뜨는		-1	١Ľ	CG87	n m	yo ,	em	-#							
na n		L L		'G		_H							LL.		CG4589		BcD	m			-10		C	G6			-8#	되					
-	CG6	CG		GC	ᆘ	-H		┢┝	+++	╓	┯╇┯	┼┼┼	┯┿┯	┍┶┶╋	-11	CG1	.50 <u>CG1</u>	sc -	-	-1999	Ξſ	lefens	e/imi	nuni	tv	enzvn	ue activ	1 8					
BG_D	002	gol (	CG H	nel	나	$-\Box$		HE		#			11,		1		Crag	м		-171#	ΗIr		CG	_	- I	Ark	TF						
	005	CG	CG 🖞		14	Н		H		#	111.						transpo	rter			-1'	Cett	0	100	~ [	~ ~		E S					
rpr	CG4	$\vdash$	na E		H	H		HF	HÌ	₽	111		H					_	J	CG4 CG	١		L	Ц,	[	chaper	r struc	rt a					
-		JCG F		2G -	1-		$\left  \right $		Ht	Ħ		HĤ			CG5254	÷	UG62.	s puess	5	.04UG			Dr	DO				<b>1</b> 8					
CG95	mbc	Eip	CG	2G -	1-	-		H	印	₽	⊞	H	ЩŢ		1				+		ſ	enzyn	ue inl	hibite	or ]			副記					
	CG1	aa	3G (		1+	$\dashv$		ĽΡ	바	₽	ŦĦŦ		睢		⊪——		_	C_			-IIr	CG13	_					- 2					
		100	CG	GC	1+	+	$\left  \right $	H	벼	┼╞	1	曲	腪		CG1390	17	CG10-	1 sut			٦ľ	0010	ľ	312.	.7	apopt	transc	r					
0004	CG8			GC	1-+	+		HF	FH				▦				0010	'C-	-		뭬		F			Bu 🕂	F+	-					

Figure 7 Treemap with nodes sized by one of the attributes

In this figure the gene with highest fold change in the ligand binding branch has a higher fold change than the gene with the highest fold change in the enzyme category. Such global level knowledge discovery becomes easy in treemaps and is less tedious than going through a list of all of the leaf nodes and then comparing the value of their attributes. The users can further specify the color of the nodes depending on particular criteria. Figure 8 shows the treemap where the color of the nodes specifies if they are basic (green) or acidic (red).

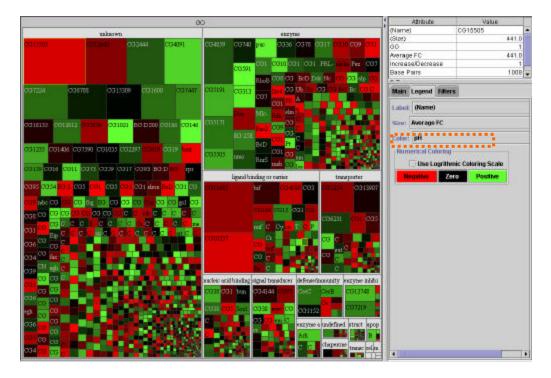


Figure 8 Treemap with nodes color by one of the attributes.

From this figure users can see that genes with highest fold change (size by attribute) in Unknown is also very acidic (Color by attribute).

# 5.2 Filtering and Dynamic queries

The control panel of the Treemap program provides the capability of dynamic queries. Dynamic queries are a visual alternative to Sequential Query Language (SQL) for querying a database. Dynamic queries are dramatically different from existing database query methods because they continuously update search results - within 100 milliseconds - as users adjust sliders or select buttons to ask simple questions, and find patterns or exceptions [4].

The control panel has sliders for each numerical attribute. Users can filter the nodes based on attributes of the leaf nodes. For example figure 9 shows the resulting treemap when all of the leaf nodes with less than 5 fold are change are filtered out.

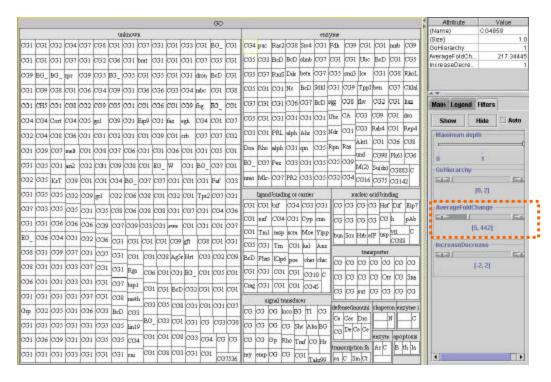


Figure 9 Treemap with nodes less than 5 fold change filtered out.

Treemap also allows any of the nodes to be set as the current node by clicking the mouse. The attributes of the current node are displayed at the right side of the screen, in the control panel. Additionally, a zooming capability is provided to allow any node in the tree to become the root node, thus obtaining more screen area for that node's descendants. For example, figure 10 shows the treemap zoomed to only enzymes.

					e	nzyme						Attribute	Value
CG4859	թաշ	Rac2	CG8946	Src42A	CG1129	Fdh	CG9364	CG1681	CG1998	mnb	CG9748	(Name) (Size) GoHierarchy AverageFoldCh	CG4859 1.0 1 217.34445
CG5191	CG3650	BcDNA_G	BcDNA_G	slmb	CG7529	CG17323	CG1837	UbcD10	BcDNA_GF	I CG10487	CG5933	IncreaseDecre	1
CG5171	CG7806	RnrS	Ddc	betaggt-I	CG7717	CG5026	smi35A	Ice	CG11058	CG8976	RhoL	Main Legend	Filters
CG5505	CG17647	CG10877	Nc	BcDNA_G	S6kII	CG11228	CG9047	TppII	ben	CG7863	CkIalpha	Size: (None) Color: (None)	
CG7402	CG10231	CG13430	CG6963	CG7221	BcDNA_G	sg	UbcD2	Ndr	Rpnó	CG5977	CG2049	Numerical Column	Zero Posi
CG5919	CG14716	CG17746	CG11956	CG12110	CG12583	CG8845	CAH1	CG3881		CG11357	dco		
CG3132	CG16973	PRL-1	alpha-Est4	Ahcy89E	CG3872	flw	CG14946	CG3534		CG1216	Rep4		
Doa	RhoBTB	alpha-Est3	CG16708	qm	CG5044	CG2272	Ras85D	Akt1		CG6612	CG8201		
EG_25E8.2	CG7656	Pez	CG3044	CG17309	CG5663	CG1473	CG9575	M(2)21AB	CG9811 	Pk61C	CG6805		
nmo	Mlc-k	CG7144	PR2	CG3752	CG 5966	kuz	CG4750	CG16701	CG7555	CG8830 CG14226	CG1764		
										CG14226		•	

Figure 10 Treemap with root set to enzyme and with nodes whose fold change less than 5 filtered out.

Users can now further size, color and filter the leaf nodes by any of its attributes. Figure 11 shows Treemap with root set to enzyme and with nodes whose fold change is less than 5 filtered out. The size-by attribute is number of base pairs and color-by attribute is pH value.

We see that among the nodes remaining after filtering, the node with highest number of base pairs (top left) is also neutral (red for acidic, green for basic and black for neutral).

				SHEVE	6				1	Atribute	Valu	
0314914	BeDNA-OB036	0016973	BarRSD	CG3977	003034	0030	H PR	1.i.	0010877	(Name)	CG14714	- and a
		1000								(Stat)	10200	4951.0
										00	-	1
										Average FC Increase/Decrease	-	6.14
					and a second					Base Pairs		4951
05505	fler	Ryad	CG1216	033919	Rub4	nap	her.	2	Rad	Dabe P MID		4901
										Main Logend File	CE .	
	C01529	C016701	Abry@9E	0017647	Ne	0025044	001330	0013480	003272	Hele Filtered Nod	es Show A	il Nodes
										🗌 Automatica	ly Hide Filtered H	lodes
											s out of 3006(55	4
		C01129			001998	007144	009047	CO12583	007221	Meximam depth		
					001000	0.01144			w121			
										u		ż
			-	002049	CO11357 60	9575 0003	0090	27 20042.6	COMME	00		
	Crigonos	007717	-							151231		Terrar
											10.25	
G7208			-	033966	CG1764 Fast		Por2	09811 00	9911 BeDtin:	and the second sec		
	033026		037402	000000						Average FC		
	000026	the								1 1		1543
				_	CO3752 CAI	IL BONA:	003132	G1489 F43	031494		(5, 442)	
				0017323						Carlo and the second second		
	002272					004730	PR2 Ptc	< 00147	Mk-k, CG333	Increase Decrease		1.000
					alito-Brt Ube	22 Berlin 1				12121		1.054.50
			0011956	54(40		007863	state state	03112	000 03517		1.2, 2]	
09748					CO1837 Nat		000 9100	001110	uno cossin	tiese Parts		
	C014226	BeDNA-OHO			CODED -			TRANK T	CORS PLAIC	5121		10000
			UbeD10	0310487		001201	Tppll	and states a	THE PROPERTY	and the second sec	9, 11995]	00000
					003	650	100	at COLL	pla 001 007	p	at a restant	
led.	- Contractory					Akt	0016			alt		
	038134	SeldT	035191	Chiston			M12	CG14	OS Der Im	E COL		1543
			00000		COID41 Doe	an	17.0		61 20 0 0		1-2.21	
							0.00	188 Pes	GP CO CO		1 and 1	

Figure 11 Treemap with root set to enzyme and with nodes whose fold change less than 5 filtered out. The treemap is sized by 'Base Pairs' and colored by 'pH' value.

### 6. Treemap strength and limitations

All static hierarchical presentations have limits as to the quantity of information they are capable of presenting on a finite display space. When these limits are reached, navigational techniques such as scrolling or panning must be used, creating the potential for loss of context [5]. Common character-based applications use a set number of lines to display the hierarchy. In graphical tree diagrams depending upon the drawing algorithm and the size of the display space, a hundred or so nodes can be adequately represented on screen without the need for panning or zooming. The number of nodes that can be displayed by a treemap can be an order of magnitude greater than traditional graphical tree diagrams. One of the treemap applications in HCIL displays 1 million nodes.

Treemap visualizations are limited to hierarchical data sets and hierarchical decompositions of categorical data [6]. As with any new visualization technique, users need some training to learn this technique.

The slice and dice algorithm, although it maintains fixed position of the nodes, usually results in thin high aspect ratio rectangles making node comparison difficult. Squarified algorithm [7] maintains low aspect ratio but results in nodes changing their position as sizes change making it difficult to find nodes consistently at one position.

### 7. Suggestions and future research

Since the success of Market Map [8] there has been renewed interest in Treemaps. Several algorithms were proposed in last few years to overcome limitations of the slice and dice algorithm (for example ordered, pivot my middle etc). A few algorithms were specialized to solve particular problems (for example Quantum treemaps). New algorithms may be developed specially for solving bio-informatics problems for example an algorithm that shows gene clustering with treemaps. Also Treemaps can be coupled with an existing visualization tool as an alternative view of data. As a web application, an applet for treemap may be created that will help users to navigate GO hierarchy.

With the present pace of developments in the field of genetics, proteomics, many more attributes of genes and proteins will be discovered and visualizing this data will be a challenging problem. New visualization techniques will be needed to organize and visualize this data. Treemap is one such powerful tool for visualizing this data. This paper discussed how treemaps could be applied to visualize gene data and its limitations.

#### 8. Acknowledgements

I would like to thank Dr Ben Shneiderman who introduced me to the interesting and challenging field of information visualization. My education in the field of HCI would have been incomplete without him. I would also like to thank Dr Eric Baehrecke for helpful discussions and comments on this manuscript.

All the demonstrations in this paper used Treemap 3.0 [9] [10] software and I would like to thank members of the Treemap team (Dr Catherine Plaisant, Jim Blowitski, Steve Betten, Jesse Grosjean) who were always helpful and supportive. I would also like to thank Agilent Technologies for supporting us on this project.

#### References

[1] Gene Ontology Consortium- www.go.org

[2] Shneiderman, B. Tree Visualization with Tree-maps: A 2-D space-filling approach. *ACM Transactions on Graphics* 11, 1 (Jan. 1992), pp. 92-99.

[3] SmartMoney- www.smartmoney.com/marketmap

[4] Shneiderman, B., Dynamic queries for visual information seeking, *IEEE Software* 11, 6 (1994), 70-77.

[5] D. Beard and J. Walker II. Navigational techniques to improve the display of large twodimensional anguage  $P_{i}$   $l_{i}$   $L_{i$ 

dimensional spaces. Behavior & Information Technology, (1990), 9(6): pp. 451-466.

[6] Johnson, B.S. "Treemaps: Visualizing Hierarchical and Categorical Data" PhD thesis, Computer Science Department, University of Maryland. (1995), p. 148.

[7] Van Wijk, J., Wetering, H. "Cushion Treemaps: Visualization of Hierarchical Information" *IEEE Symposium on Information Visualization*, (1999), pp. 73-78

[8] 2001 National Magazine award for Best Interactive Design

[9] http://www.cs.umd.edu/hcil/treemap3/

[10] http://www.cs.umd.edu/hcil/treemaps/