The interpretation and utility of three cohesion metrics for object-oriented design

Article in ACM Transactions on Software Engineering and Methodology · April 2006
DOI: 10.1145/1131421.1131422 · Source: DBLP

Stephen Swift
Brunel University London

Jason Crampton
Royal Holloway, University of London

The Interpretation and Utility of Three Cohesion Metrics for Object-Oriented Design

Steve Counsell, Stephen Swift, and Jason Crampton

ACM TOSEM, April 2006

2006. 7. 26

Taehoon Song

Contents

- Introduction
- Motivation and related work
- Preliminaries
- Cohesion metrics
- Empirical results
- Conclusion and further work
- Discussion

Introduction (1/2)

- Definition of cohesion
  - Is regard *a* **class as cohesive** *if the* ***methods of the class*** ***use*** *the* ***same set of parameter types*** **[1]**
- Metrics for measuring cohesion
  - Cohesion among methods in a class metric (CAMC) [1]
  - Normalised hamming distance metric (NHD) [2]
  - Scaled NHD metric

[1] Bansiya, J., Etzkorn, L., Davis, C., and Li, W. 1999. A class cohesion metric for object-oriented designs. J. Object-Oriented Program. 11,8, 47-52.

[2] Counsell, S., Mendes, F., Swift, S., and Tucker, A. 2002. Evaluation of an object-oriented cohesion metric through Haming distances. Tech. Rep.BBKCS-02-10, Birkbeck College, University of London, UK.

Introduction (2/2)

- Purpose of this article
  - Rigorous **mathematical analysis**
    - Determine ***whether these metrics*** have any qualitative meaning given the definition of cohesion above
    - Determine ***what values of these metrics*** should be considered to represent a cohesive class

Motivation and related work (1/2)

- Mathematical comparison of the properties of cohesion metrics
  - Is an **under researched area**
- Identifying common failings or properties of cohesion metrics
  - Informs our understanding of OO system, OO language and their different traits

Examination and scrutiny of **current** cohesion metrics

Motivation and related work (2/2)

- Object-oriented paradigm
  - Notion of class cohesion has superceded that of module of cohesion
    - Lack of cohesion of methods metric (LCOM) [3]
      - On the assumption that a class is cohesive if the same instance variables appear in most or all of the methods in a class
        - Values produced by the metric are difficult to interpret and give little insight into the nature of the class
        - The metric is an implementation metric
          - ✓Measure of cohesion is required earlier in the development process (at **design time**)

[3] Chidamber, S. and Kemerer, C. 1994. A metrics suite for object oriented design. IEEE Trans. Soft. Eng. 20, 6, 467-493.

Preliminaries (1/4)

- Three systems
  - Represent a variety of different application domains
    - Edge (graph editor : 30.8 KNCSL, 80 classes)
    - Rocket (compiler : 32.4 KNCSL, 322 classes)
    - Et++ (user interface framework : 56.3 KNCSL, 508 classes)

☆ KNCSL : thousand noncomment source lines

- Using classes randomly chosen from these systems

Preliminaries (2/4)

- Notion of an entity relational system (ERS)
  - Provides a mapping from the real world attribute of the entity being measured to values in the empirical world
    - Assigns a measure of the similarity between the parameter types of the methods for each class
      - A class X *is more cohesive than* class Y if this function returns a higher value for X when there is greater sharing of parameters between the methods of a class
  - Distinguish between the cohesiveness of *n* classes using the same metric

Preliminaries (3/4)

- Notation
  - (i, j)th entry of a matrix M : m_{ij}
  - Given class : C
    - Methods : k *(row vector)*
    - Parameter type list : L *(column vector)*
      - Length of L : l
  - Parameter occurrence matrix O $(1 \leq i \leq k,\ 1 \leq j \leq l)$

$$O_{ij} = \begin{cases} 1 & \text{if the } j\text{th data type occurs as a parameter in the } i\text{th method} \\ 0 & \text{otherwise} \end{cases}$$

Preliminaries (4/4)

$$\text{Alert}\quad \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

■ Notation (Cont'd)

```
Alert(AlertType, byte, *text=0, Bitmap *bm=0);
~Alert();
VObject *DoCreateDialog();
int Show(char *fmt);
int ShowV(char *fmt, va_list ap);
class Menu *GetMenu();
void InspectorId(char *buf, int sz);
```

(a) **Methods**

Binary *k*l* matrix

- *i*th row : parameter occurrence vector (for method *i*)
  - Indicates the presence of data types in the *i*th method

$$\text{Alert}\quad \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

	va_list	int
Alert	0	0
~Alert	0	0
DoCreateDialog	0	0
Show	0	0
ShowV	1	0
GetMenu	0	0
InspectorId	0	1

(b) **Parameter occurrence matrix**

$$r_i = \sum_{j=1}^{l} o_{ij}, \quad c_j = \sum_{i=1}^{k} o_{ij}, \quad \sigma = \sum_{i=1}^{k}\sum_{j=1}^{l} o_{ij}$$

Cohesion metrics (1/6)

- CAMC
  - Can be evaluated at design time (↔ *LCOM*)
  - Is the average of the entries in the parameter occurrence matrix
  - Formulation

Cohesion indicator : **0.35**

$$\mathrm{CAMC}(C) = \frac{1}{kl}\sum_{i=1}^{k}\sum_{j=1}^{l} o_{ij} = \frac{\sigma}{\boldsymbol{kl}}$$

k : methods
l : length of parameter type list

$$\mathrm{CAMC}(\mathrm{Alert}) = \frac{1}{7*6}(3+0+0+1+2+0+2) = 8/42 \approx 0.19$$

Cohesion metrics (2/6)

- CAMC (Cont'd)
  - CAMCs
    - Includes the "**self**" parameter type in the parameter occurrence matrix
      - Append a **column of 1s** to the parameter occurrence matrix forming the (l+1)th column

To provide a value for the CAMC metric *when no methods of the class had any parameters*

$$\text{CAMCs}(C) = \frac{\sigma + k}{K(l+1)}$$

$$\text{CAMC}(C) = \frac{\sigma}{kl}$$

$$\text{CAMCs}(\text{Alert}) = \frac{8+7}{7*(6+1)} = 15/49 \approx 0.31$$

Cohesion metrics (3/6)

- NHD
  - Measures agreement between rows in a binary matrix
    - Alternative measure of the cohesion in the sense computed by the CAMC metric
  - Parameter agreement matrix A

$$A = \begin{cases} a_{ij} & \text{if the parameter agreement is between methods } i \text{ and } j \ (1 \leq j < i \leq k) \\ 0 & \text{otherwise} \end{cases}$$

Cohesion metrics (4/6)

- NHD (Cont'd)
  - Parameter agreement matrix
    - Lower triangular square matrix of dimension k-1

Cohesion indicator : **0.5**

(c) Formulation

$$\mathrm{NHD}(C) = \frac{1}{l\binom{k}{2}} \sum_{j=1}^{k-1} \sum_{i=j+1}^{k} a_{ij} = \frac{2}{lk(k-1)} \sum_{j=1}^{k-1} \sum_{i=j+1}^{k} a_{ij}$$

$$\mathrm{NHD}(\mathrm{Alert}) = \frac{2}{6*7*(7-1)} * \boxed{84} = 168/252 \approx 0.67$$

O	AlertType	byte	Bitmap	char	Va_list	int
Alert	1	1	1	0	0	0
Show	0	0	0	1	0	0
InspectorId	0	0	0	1	0	1

(a) Parameter occurrence matrix

O	Alert	~Alert	DoCreateDialog	Show	ShowV	GetMenu
	3					
~Alert	3 / 3	6				
DoCreateDialog	2 / 3	5 / 6	5			
Show	1 / 2	4 / 5	4 / 5	5		
ShowV	3 / 1	6 / 4	6 / 4	5 / 5	4	
GetMenu	1 / 3	4 / 6	4 / 6	5 / 5	4 / 4	4
InspectorId	13 / 1	25 / 4	19 / 4	15 / 5	8 / 4	4 / 4
	13	25	19	15	8	4

(b) **Parameter agreement matrix**

Cohesion metrics (5/6)

- NHD (Cont'd)

$$\mathrm{NHD}(C) = \frac{1}{l\binom{k}{2}}\sum_{j=1}^{k-1}\sum_{i=j+1}^{k} a_{ij} = \frac{2}{lk(k-1)}\boxed{\sum_{j=1}^{k-1}\sum_{i=j+1}^{k} a_{ij}}$$

  - Can distinguish between different parameter occurrence matrices with the same number of 1s
    - $\mathrm{NHD}_{min} \le \mathrm{NHD} \le \mathrm{NHD}_{max}$ $(l \le \sigma \le kl)$

$\sigma = 8$
$k = 7$
$l = 6$

$$\mathrm{NHD}_{min} = 1 - \frac{2}{lk*(k-1)}(q(d+1)(k-d-1)+(l-q)d(k-d))$$

$$\mathrm{NHD}_{max} = 1 - \frac{2}{lk*(k-1)}(r+1)(k-r-1)+(l-c$$

$d = \left\lfloor \frac{\sigma}{l} \right\rfloor$, σ

O	t_1	t_2	t_3	t_4	t_5	t_6
m_1	1	1	1	1	1	1
m_2	1	1	0	0	0	0
m_3	0	0	0	0	0	0
m_4	0	0	0	0	0	0
m_5	0	0	0	0	0	0

O	t_1	t_2	t_3	t_4	t_5	t_6
m_1	1	1	1	1	1	1
m_2	1	0	0	0	0	0
m_3	1	0	0			
m_4	0	0	0			
m_5	0	0	0			

2					
0	4				
0	4	6			
0	4	6	6		
0	4	6	6	6	
0	4	6	6	6	6
2	20	24	18	12	6

Parameter agreement matrix

$\boxed{82 < 84}$

1					
1	6				
0	5	5			
0	5	5	6		
0	5	5	6	6	
0	5	5	6	6	6
2	26	20	18	12	6

Cohesion metrics (6/6)

- Scaled NHD
  - Uses of the fact that both ends of the range of values for NHD
  - Represents how close the NHD metric
    - is to the maximum value of NHD compared to the minimum value

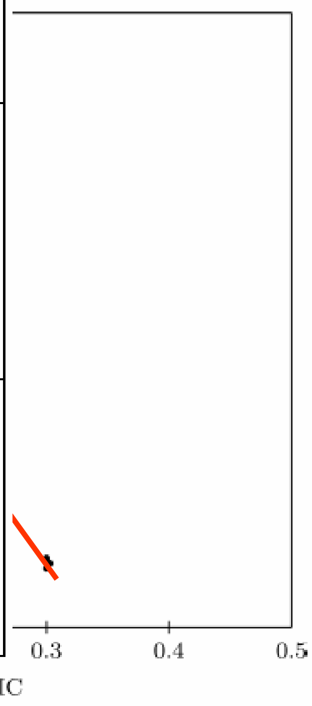
$$SNHD = \begin{cases} 0 & \text{if } NHD_{min} = NHD_{max} \text{ and } \sigma < kl \\ 1 & \text{if } \sigma = kl \\ 2\left(\dfrac{NHD - NHD_{min}}{NHD_{max} - NHD_{min}}\right) - 1 & \text{otherwise} \end{cases}$$

Empirical results (1/2)

- Evaluation of cohesion metrics

System	Class	k	l	$CAMC_s$	CAMC	NHD_s	NHD	$SNHD_s$	SNHD
Et++	Alert	7	6	0.306	0.190	0.714	0.667	1.000	1.000
	ApplDialog	4	3	0.438	0.250	0.625	0.500	1.000	0.000
	BagItem	11	4	0.309	0.136	0.804	0.755	1.000	1.000
	Dialog	15	6	0.248	0.122	0.810	0.778	0.830	−0.586
	CycleItem	14	11	0.202	0.130	0.797	0.778	0.512	−0.451
	BitMap	22	10	0.169	0.086	0.856	0.842	0.757	−0.555
	Assoc	11	3	0.409	0.212	0.773	0.697	1.000	1.000
Rocket	Arc	5	2	0.467	0.200	0.733	0.600	1.000	0.000
	ArcList	9	3	0.389	0.185	0.764	0.685	1.000	1.000
	CallGraph	11	4	0.273	0.091	0.855	0.818	1.000	0.000
	DDGArcTypeList	9	3	0.389	0.185	0.764	0.685	1.000	1.000
	DDGNNodePtrList	10	3	0.475	0.300	0.661	0.548	0.227	−0.835
	DataType	20	5	0.225	0.070	0.887	0.864	0.989	−1.000
	DeclaratorPtrList	11	3	0.477	0.303	0.664	0.552	0.177	−0.875
Edge	null_dummy	7	4	0.343	0.179	0.733	0.667	1.000	0.000
	constr_descriptor	8	9	0.225	0.139	0.757	0.730	1.000	0.000
	constr_queue	8	8	0.292	0.203	0.687	0.647	0.344	−0.807
	constr_manager	17	8	0.229	0.132	0.827	0.805	0.773	0.206
	gne_default	14	3	0.393	0.190	0.775	0.700	0.868	0.013
	elist	10	2	0.533	0.300	0.704	0.556	0.521	−0.490
	intersect	21	4	0.314	0.143	0.800	0.750	0.758	−0.750

CAM
CAM
NHD
NHD
SNH



Empirical results (2/2)

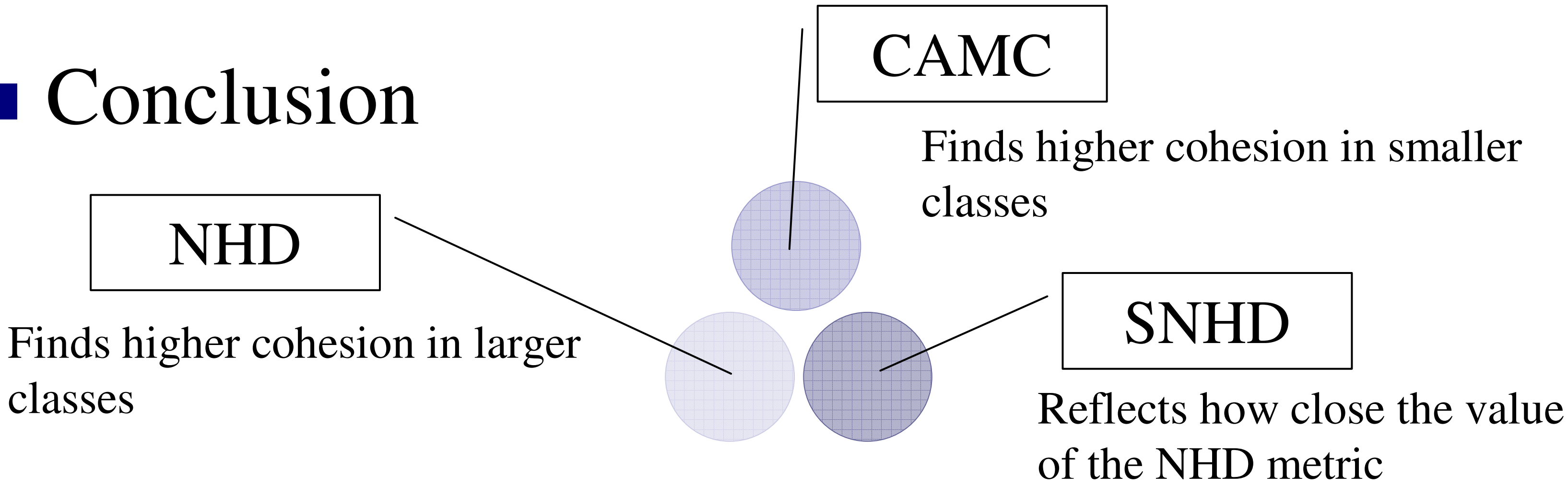
- Cross comparison of the three metrics
  - Correlations between the three cohesion metrics

Comparison	Pearson's	Spearman's	Kendall's	***Mean***
LCOM versus SNHD	-0.458	-0.425	-0.337	***-0.407***
LCOM versus CAMC	-0.540	-.0239	-0.156	***-0.312***

- Table shows evidence of correlation between design-time metrics (such as CAMC and SNHD) and code-based metrics (such as LCOM)
  - SNHD metric produces higher results than CAMC

Conclusion and further work

- Conclusion



- Further work
  - Undertake a more formal and extensive analysis of the SNHD metric
  - Conduct more extensive tests on whole systems
  - Establish significant values of the SNHD metric

Discussion

- Discussion
  - Need to make a best definition of cohesion
    - What cohesion means in object-oriented software in light of the matrices O_{max} and O_{min}
  - Need to adjust more system
    - Exclude prior knowledge (Edge, rocket, et++)
    - Use more various number of classes
      - Threat to scalability of the results
    - Choose classes in more OO application types
  - Need to interplay between cohesion and OO coupling

Background

- Original definition of the metric in OO sense
  - Calculates cohesion according to use of class attributes in the methods of a class
  - Is based on the principle that an instance variable occurring in many methods of a class
    - Causes that class to be more cohesive than one where the same variable is used in very few methods of the class
- High cohesion
  - Reflects using of development technique known to produce robust and maintainable code

Root of cohesion (1/2)

- Procedural programming viewpoint
  - Modules were the key elements by which cohesion was measured
    - Inter-module metrics (Stevens et al. [1974])
    - Seven point ordinal scale for component cohesion (Yourdon and constantine [1979])
      - Functional cohesion
        - Module perform a single well-defined function
      - Coincidental cohesion
        - Module perform more than on function, and that those functions were unrelated

Root of cohesion (2/2)

- Structured paradigm
  - Informal definition of cohesion (Lakhotia [1993])
    - Was built on the basis of sound programmer practice and experience
    - Was underpinned work on measuring module cohesion

Cohesion metrics (1/2)

- CAMC
  - Interpreting
    - Can't distinguish between the cohesion of different matrices with the same value of σ
    - Has fundamentally flawed about using 0.35 as a threshold for an indicator
    - Is likely to find smaller classes more cohesive, irrespective of their actual properties

kl	k	l	σ	CAMC	σ_s	CAMC_s
6	2	3	3	0.500	4	0.667
	3	2	2	0.333	4	0.667
12	2	6	6	0.500	7	0.583
	3	4	4	0.333	6	0.500
	4	3	3	0.250	6	0.500
	6	2	2	0.167	7	0.583
24	2	12	12	0.500	13	0.542
	3	8	8	0.330	10	0.417
	4	6	6	0.250	9	0.375
	6	4	4	0.167	9	0.375
	8	3	3	0.125	10	0.417
	12	2	2	0.083	13	0.542
36	2	18	18	0.500	19	0.528
	3	12	12	0.333	14	0.389
	4	9	9	0.250	12	0.333
	6	6	6	0.167	11	0.306
	9	4	4	0.111	12	0.333
	12	3	3	0.083	14	0.389
	18	2	2	0.056	19	0.528

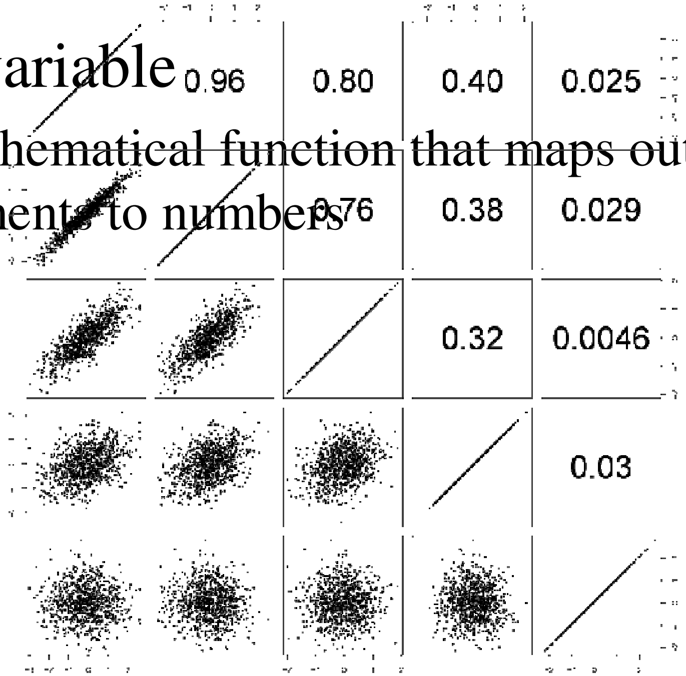
<Minimum values of CAMC and CAMCs >

Cohesion metrics (2/2)

- Hamming distance (HD) metric
  - Provides a measure of disagreement between rows in a binary matrix informally (Counsell et al. [2001])
- NHD
  - Interpretation
    - Suggest that a class for which the NHD metric is more than 0.5 should be considered cohesive (Counsell et al. [2002])
    - Must reconsider carefully what we mean by cohesion
      - It is questionable whether this is satisfactory behavior for a cohesion metric, as small classes are generally regarded as being more cohesive than large ones

Correlation (1/5)

- Correlation (correlation coefficient)
  - Indicates the strength and direction of a linear relationship between two random variables
    - Random variable
      - is a mathematical function that maps outcomes of random experiments to numbers



Correlation (2/5)

- Pairwise independence
  - Collection of random variables is a set of random variables any two of which are independent
    - Suppose X,Y and Z have the following joint probability distribution

$$(X, Y, Z) = \begin{cases} (0,0,0) & \text{with probability } 1/4, \\ (0,1,1) & \text{with probability } 1/4, \\ (1,0,1) & \text{with probability } 1/4, \\ (1,1,0) & \text{with probability } 1/4. \end{cases}$$

      - X-Y, X-Z, Y-Z are independent
      - X,Y,Z are not independent
        - Mod 2 sum of the other two is completely determined by other two

Correlation (3/5)

- Pearson's correlation coefficient
  - Is defined only if both of the standard deviations are finite and both of them are nonzero
    - Is a corollary of the Cauchy-Schwarz inequality that the correlation cannot exceed 1 in absolute value
  - Is 1 in the case of an increasing linear relationship, −1 in the case of a decreasing linear relationship
    - The closer the coefficient is to either −1 or 1, the stronger the correlation between the variables.
  - If the variables are independent then the correlation is 0, but the converse is not true because the correlation coefficient detects only linear dependencies between two variables.

Correlation (4/5)

- Spearman's correlation coefficient
  - Is a non-parametric measure of correlation
    - Assesses how well an arbitrary monotonic function could describe the relationship between two variables, without making any assumptions about the frequency distribution of the variables

$$\rho = 1 - \frac{6 \sum D^2}{N(N^2 - 1)}$$

D = the difference between the ranks of corresponding values of X and Y, and
N = the number of pairs of values
Spearman's rank correlation coefficient ***is equivalent to*** *Pearson correlation on ranks*

Correlation (5/5)

- Kendall's correlation coefficient

$$\tau = \frac{2P}{\frac{1}{2}n(n-1)} - 1$$

relationships between different

e same set of items

τ has the following properties:

- If the agreement between the two rankings is perfect, the coefficient has value 1
- If the disagreement between the two rankings is perfect, the coefficient has value -1
- If the rankings are independent, the coefficient has value 0

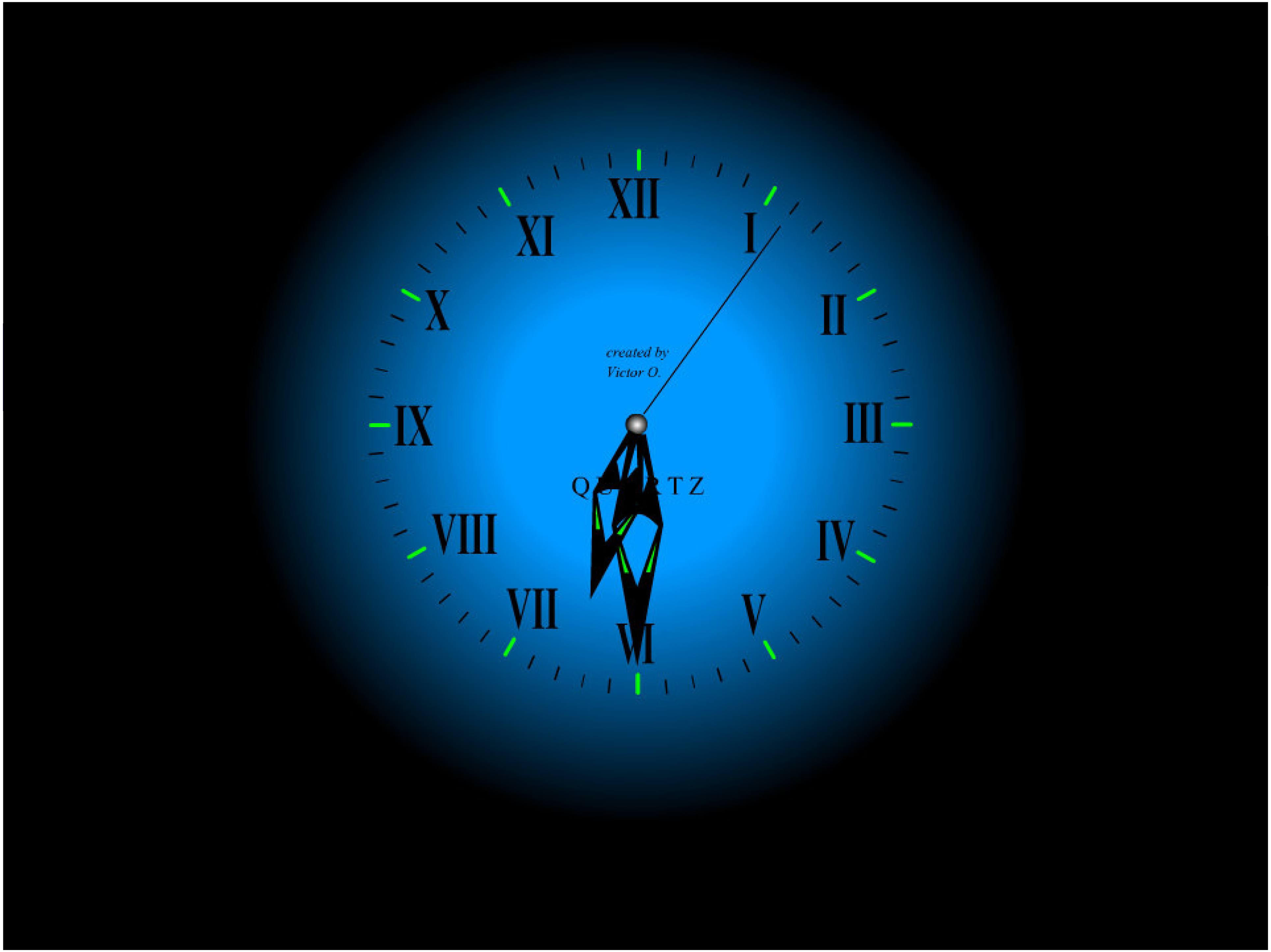
  - Deals with measuring correspondence between two rankings and assessing the significance of this

:e

P = 5+4+5+4+3+1+0+0=22

$$\tau = \frac{44}{28} - 1 = 0.57$$

Person	A	B	C	D	E	F	G	H
Rank by height	1	2	3	4	5	6	7	8
Rank by weight	3	4	1	2	5	7	8	6


XII
I
II
III
IV
V
VI
VII
VIII
IX
X
XI
created by
Victor O.