目前的相关文献包括：

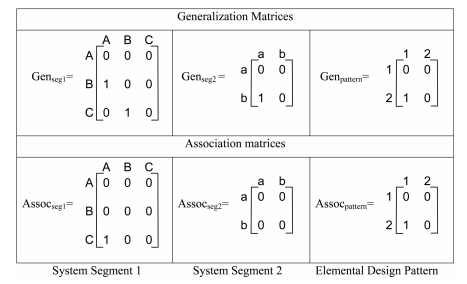
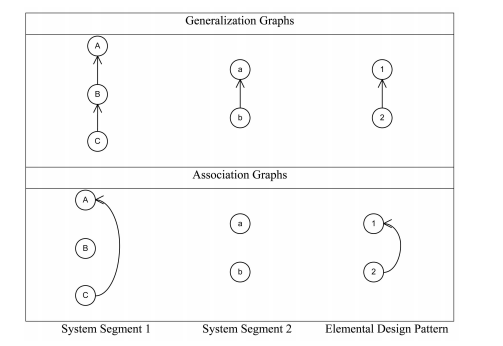
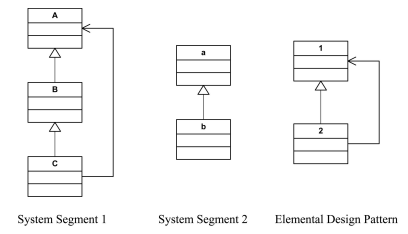
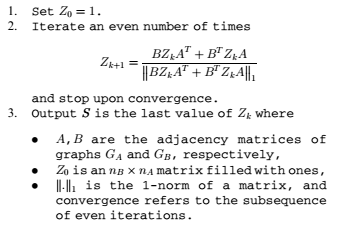
1. 设计模式相关
2. 词向量相关
3. 变异算子
4. 变异算子与真实错误

## 一、设计模式

**[1]Design Pattern DetectionUsing Similarity Scoring**

**[2]Application of Graph Theory to OO Software Engineering**

**使用图相似度的方法 进行 设计模式的匹配**



from numpy import \*

import numpy as np

from numpy import linalg as LA

np.seterr(divide='ignore', invalid='ignore')

def getnext(B,A,ZK):

temp= B.dot(ZK).dot(A.T)+B.T.dot(ZK).dot(A)

# print LA.norm(temp,1)

if LA.norm(temp,1)==0:

return temp

else:

ZKN=temp/LA.norm(temp,1)

return ZKN

def similar(B,A):

Z0 = ones((B.shape[0], A.shape[1]))

Z=Z0

ZN=Z0

while(True):

ZN=getnext(B,A,Z)

if (ZN==Z).all():

return ZN

Z=ZN

GP=array( [ (0,0), (1,0) ] )

AP=array([(0,0),(1,0)])

GB=array( [ (0,0,0), (1,0,0),(0,1,0) ] )

AB=array([(0,0,0),(0,0,0),(1,0,0)])

print (similar(B=GB,A=GP)+similar(B=AB,A=AP))

GB=array([(0,0),(1,0)])

AB=array([(0,0),(0,0)])

print (similar(B=GB,A=GP)+similar(B=AB,A=AP))

## 词向量