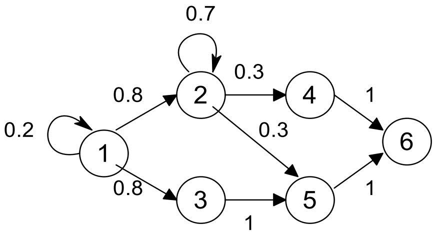
Appendix 2: Sample examples

Example A-1: Composition graph aggregation without conditional pattern for the first member of sPop population

Composition graph



Composition graph with 6 abstract web services

$cComp_1=(cc,Q)_1$

$cComp_1.cc$ ={341,797,890,1635,1936,2130}

$cComp_1.Q_1$=[0.7900,0.5684,0.0190]

$cComp_1.Q_2$=[0.8800,0.0349,0.0046]

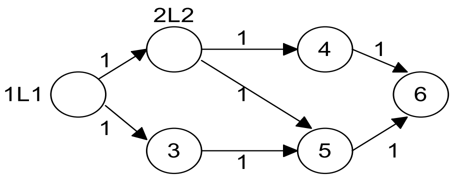
$cComp_1.Q_3$=[0.8500,0.0249,0.0027]

$cComp_1.Q_4$=[0.9100,0.0782,0.0021]

$cComp_1.Q_5$=[1,0.0156,2.4153e-04]

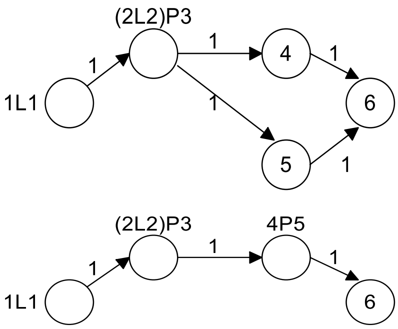
$cComp_1.Q_6$=[0.9600,0.0205,0.0103]

#web service	Q=[availability,response-time,cost]
1	[0.7900,0.5684,0.0190]
2	[0.8800,0.0349,0.0046]
3	[0.8500,0.0249,0.0027]
4	[0.9100,0.0782,0.0021]
5	[1,0.0156,2.4153e-04]
6	[0.9600,0.0205,0.0103]



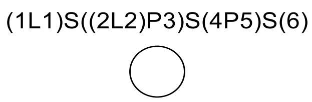
First step: removing the loop pattern (nodes 1 and 2)

#web service	Q=[availability,response-time,cost]
1	[0.7506,0.7105,0.0238]
2	[0.6875,0.1164,0.0152]
3	[0.8500,0.0249,0.0027]
4	[0.9100,0.0782,0.0021]
5	[1,0.0156,2.4153e-04]
6	[0.9600,0.0205,0.0103



Second step: removing the parallel pattern

#web service	Q=[availability,response-time,cost]
1	[0.7506,0.7105,0.0238]
2	[0.5844,0.1164,0.0178]
4	[0.9100,0.0782,0.0023]
6	[0.9600,0.0205,0.0103]



Third step: removing the sequence pattern and produce summary node

#web service	Q=[availability,response-time,cost]
1	[0.3832,0.9256,0.0542]

Q'_1= [0.3832,0.9256,0.0542]

$sPop_1=(cComp,Q')_1$

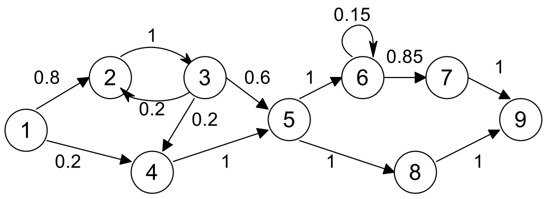
Example A-2: Composition graph aggregation with conditional pattern for the first member of sPop population

Composition graph

$cComp_1=(cc,Q)_1$

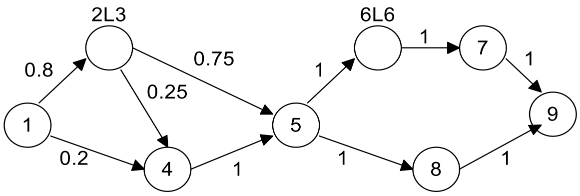
$cComp_1.cc$={10,401,664,1049,1336,1445,1808,2074,240
9}

#web service	Q=[availability,response-time,cost]
1	[0.9600,0.0187,0.0101]



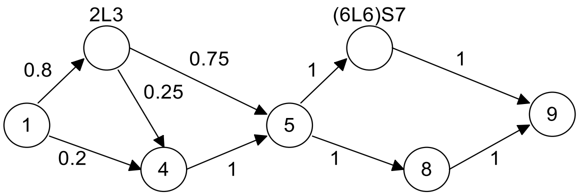
Composition graph with 9 abstract web services

2	[0.7200,0.2384,4.8305e-04]
3	[0.9300,0.0121,0.0022]
4	[1,0.0392,0.0062]
5	[0.8800,0.1705,0.1151]
6	[0.8300,0.0154,7.2458e-04]
7	[0.9600,0.0397,0.0121]
8	[0.9300,0.4658,0.3843]
9	[0.9200,0.0401,0.0020]



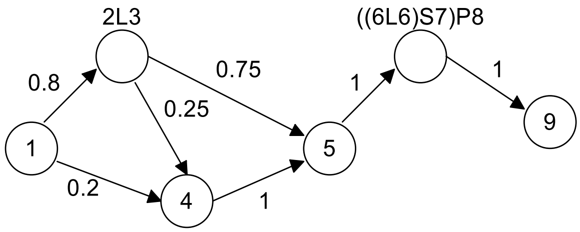
First step: removing the loop patterns

#web service	Q=[availability,response-time,cost]
1	[0.9600,0.0187,0.0101]
2	[0.6185,0.3131,0.0033]
4	[1,0.0392,0.0062]
5	[0.8800,0.1705,0.1151]
6	[0.8058,0.0328,8.5244e-04]
7	[0.9600,0.0397,0.0121]
8	[0.9300,0.4658,0.3843]
9	[0.9200,0.0401,0.0020]



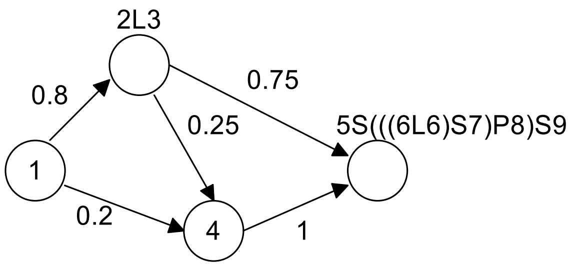
Second step: removing the sequence patterns

#web service	Q=[availability,response-time,cost]
1	[0.9600,0.0187,0.0101]
2	[0.6185,0.3131,0.0033]
4	[1,0.0392,0.0062]
5	[0.8800,0.1705,0.1151]
6	[0.7736,0.0725,0.0129]
8	[0.9300,0.4658,0.3843]
9	[0.9200,0.0401,0.0020]



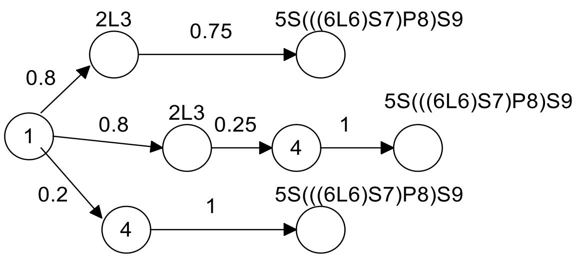
Third step: removing the parallel patterns

#web service	Q=[availability,response-time,cost]
1	[0.9600,0.0187,0.0101]
2	[0.6185,0.3131,0.0033]
4	[1,0.0392,0.0062]
5	[0.8800,0.1705,0.1151]
6	[0.7194,0.4658,0.3972]
9	[0.9200,0.0401,0.0020]



Fourth step: removing the sequence patterns

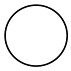
#web service	Q=[availability,response-time,cost]
1	[0.9600,0.0187,0.0101]
2	[0.6185,0.3131,0.0033]
4	[1,0.0392,0.0062]
5	[0.5825,0.6765,0.5143]



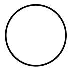
$Path_1=\{1,2,5\}$	$Prob_1=0.6$
$Path_2=\{1,2,4,5\}$	$Prob_2=0.2$
$Path_3=\{1,4,5\}$	$Prob_3=0.2$

Fifth step: extracting the path

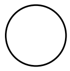
1**S**(2L3)**S**(5S(((6L6)S7)P8)S9)



1**S**(2L3)**S**4**S**(5S(((6L6)S7)P8)S9)



1**S**4**S**(5S(((6L6)S7)P8)S9)



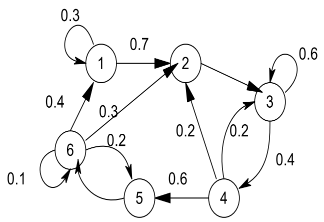
1	$Path_1$={1,2,5}	Q'_1 =[0.3458,1.0083,0.5277]
1	$Path_2$={1,2,4,5}	Q'_1 =[0.3458,1.0475,0.5339]
1	$Path_3$={1,4,5}	Q'_1 =[0.5592,0.7344,0.5306]

Sixth step: removing the sequence pattern and produce summary node for each path

Example A-3. An example of aggregating the loop pattern with 6 services

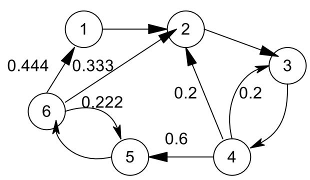
Composition graph

Find loops in order from the longest to the smallest using Algorithm A-4



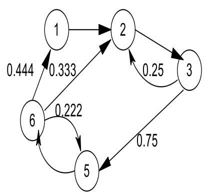
{1,2,3,4,5,6}
{2,3,4,5,6}
{2,3,4}
{5,6}
{3,4}
{6}
{1}
{3}

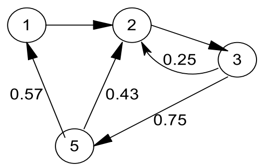
Composition graph by removing loops of length 1

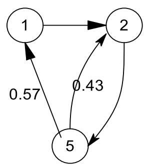


{1,2,3,4,5,6}
{2,3,4,5,6}
{2,3,4}
{5,6}
{3,4}

Composition graph by removing loops of length 2



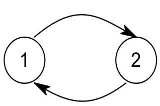


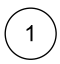


{1,2,3,5,6}
{2,3,5,6}
{2,3}
{5,6}

{1,2,3,5}
{2,3,5}
{2,3}

{1,2,5}
{2,5}

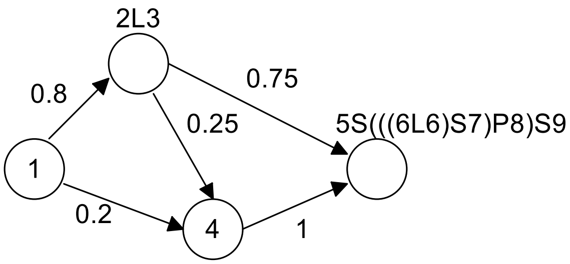




{1,2} {1}

Example A-4: An example of extracting paths of a composite graph with 6 nodes after running Algorithm A-5.

This example shows the contents of the Array and path arrays in each step of the running the Algorithm A-5 for extracting the path of the below composition graph



Array values								
1	2	3	4	5	6	7	8	9
[1]	[1]	[1,2]	[1,2]	[1,4]	[1,4]	[1,2,4]	[1,2,4]	[1,2,5]
	[1,2]	[1,4]	[1,4]	[1,2,4]	[1,2,4]	[1,2,5]	[1,2,5]	[1,4,5]
	[1,4]		[1,2,4]	[1,2,5]	[1,2,5]	[1,4,5]	[1,4,5]	[1,2,4,5]
			[1,2,5]		[1,4,5]		[1,2,4,5]	
Line2	Lines 5,6	Line 11	Line 5,6	Line 11	Line 5,6	Line 11	Line 5,6	Line 11

Path values	Array values	Path values	Array values	Path values	Array values
[10]	[11]	[12]	[13]	[14]	[15]
[1,2,5]	[1,4,5]	[1,2,5]	[1,2,4,5]	[1,2,5]	
	[1,2,4,5]	[1,4,5]		[1,4,5]	
				[1,2,4,5]	
Line 8	Line 11	Line 8	Line 11	Line 8	Line 11

Example A-5. This example shows the values of the variables used in Algorithm 5 (*heuristic* prioritizing solution function) step by step.

Suppose that the random population selected at the beginning of the algorithm consists of four (Npop'=4) solutions: sPop'={$(cComp,Q')_1$, $(cComp,Q')_2$, $(cComp,Q')_3$, $(cComp,Q')_4$}

where each solution in sPop' includes a pair of cComp (a pair of candidate index and corresponding feature) and Q' (summary node). In the priority function, we only deal with summary nodes (set Q'). We know that each Q'_i contains three features q'_{i1}, q'_{i2} , q'_{i3}. If the values of these features and its F value for sPop' members are as follows:

Q'_1=(0.79,0.23,0.01) F_1=0.8
Q'_2=(0.82,0.42,0.03) F_2=0.2
Q'_3=(0.52,0.1,0.1) F_3=0.4
Q'_4=(0.67,0.3,0.008) F_4=1

After running lines 2 and 3 of Algorithm 4, sF and sortedF are displayed as below.

sF={0.8,0.2,0.4,1}
sortedsF=(F_2,F_3,F_1,F_4)=(0.2, 0.4,0.8,1)

After running Line 4 of Algorithm 4 and calling the priority function, the sorted1 set is sorted descending on q'_{i1}, and the sorted2 and sorted3 sets are sorted ascending on q'_{i2} and q'_{i3}. The sum of the average position of each q'_{ik} and the position of each F_i is determined as the final priority of the solutions in the sortedPop set.

sorted1={q'_{21}, q'_{11}, q'_{41}, q'_{31}}
sorted2={q'_{32}, q'_{12}, q'_{42}, q'_{22}}
sorted3={q'_{43}, q'_{13}, q'_{23}, q'_{33}}

$sortedPop_1$=[(2+2+2)/3]+3=5, /* for Q'_1,
$sortedPop_2$=[(1+4+3)]+1=3.7, /* for Q'_2
$sortedPop_3$=[(4+1+4)/3]+2=5, /* for Q'3
$sortedPop_4$=[(3+3+1)/3]+4=6.3 /* for Q'_4
newsPop'=($(cComp,Q')_2$, $(cComp,Q')_1$, $(cComp,Q')_3$, $(cComp,Q')_4$)

Example A-6: This example shows the values of the variables used in Algorithm 6 (discretization function) step by step.

Suppose that $3 \in C(As_1)$ ،$600 \in C(As_2)$،$1400 \in C(As_3)$ ،$2300 \in C(As_4)$ (see Definition 2) consider for abstract web services As_1, As_2, As_3, As_4 respectively. and As4 (see Relation 1) is considered and itomega=1. So, we have:

$(w_{\omega})_1$.cComp={(3, (0.82,0.2,0.032)), (600, (0.75,0.1,0.001)), (1400,(0.68,0.15,0.003), (2300, (0.78,0.3,0.002))

where 3,600,1400,2300 are candidate numbers from $C(As_1)$, $C(As_2)$, $C(As_3)$, $C(As_4)$, respectively for As_1, As_2, As_3, As_4

and (0.82,0.2,0.032), (0.75,0.1,0.001) ,(0.68,0.15,0.003), (0.78,0.3,0.002) are (Q_{1-4}) the quality values for these candidate numbers.

In the same way, for alpha, beta and delta wolves (solutions) we also have:
w_{α}.cComp ={(50, (0.92,0.32,0.005)), (400,(0.73,0.4,0.006)), (1100, (0.88,0.2,0.002)), (2100,(0.96,0.5,0.005))}
w_{β}.cComp={(20,(0.88,0.2,0.002)), (740,(0.64,0.35,0.04)), (1800, (0.93,0.1,0.002)),(2050,(0.62,0.1,0.005))}
w_{δ}.cComp ={(210,(0.96,0.5,0.07)), (531,(0.62,0.01,0.006)), (1920,(0.89,0.18,0.02)), (2150,(0.82,0.25,0.007))}

The quality values of the new omega wolf in continuous space are as follows.
sNewOmeagContinues ={(0.87,0.23,0.006),(0.59,0.018,0.008),(0.91,0.3,0.0012),(0.71,0.3,0.008)}

Now we want the candidate index and quality values of the new omega wolf in the discrete space.
$sNewOmega_1$.cComp={ (?,?) , (?,?) , (?,?), (?,?)}

So, we should determine the candidate index and the corresponding feature for each dimension of the sNewOmega which is specified as (?,?).
Based on the proposed discretization function, the quality values of each dimension of sNewOmeagContinues are compared with each dimension of the current wolves, alpha, beta and delta, and if dominated by any of these wolves, the dominant wolf candidate index is selected. In this example, the quality values of the first dimension of the new wolf in the continuous space are (0.87, 0.23, 0.006), which are first compared with the quality values of the first dimension of the current wolf (0.82, 0.2, 0.032) and because the current wolf does not dominate it, so it is compared to alpha wolf (0.92, 0.32, 0.005), and since it is not dominated by the alpha wolf too, compared to the beta wolf. These values in beta wolf are (0.88, 0.2, 0.002), so the beta wolf (with candidate index 20) dominates the new wolf in the first dimension, and the candidate index and the features corresponding to it are determined for the first dimension of the new wolf in the discrete space:

$sNewOmega_1$.cComp={ (20,(0.88,0.2,0.002)), (?,?) , (?,?), (?,?)}

Now the quality values of the second dimension of the new wolf (0.59, 0.018, 0.008) with the quality values of the second dimension of the current wolves (0.75, 0.1, 0.001), alpha (0.73, 0.4, 0.006), beta (0.93, 0.1, 0.002) and delta (0.62, 0.01, 0.006) is compared and Delta wolf dominates it. Therefore, the candidate index and the features corresponding to it for the second dimension of the new wolf are also determined.
$sNewOmega_1$.cComp={ (20,(0.88,0.2,0.002)), (531,(0.62,0.01,0.006)), (?,?), (?,?)}

Now the quality values of the third dimension of the new wolf (0.91, 0.3, 0.0012) with the quality values of the third dimension of the current wolves (0.68, 0.15, 0.003), alpha (0.88, 0.2, 0.002), beta (0.64, 0.35, 0.04) and Delta (0.89, 0.18, 0.02) is compared and none of them dominates the new wolf. So, all available candidate indexes for this web service are reviewed. The first candidate that could dominate the new wolf is selected. For example, the quality values of the 1990 candidate dominated the quality values of this dimension of the new wolf.
$sNewOmega_1$.cComp={ (20,(0.88,0.2,0.002)), (531,(0.62,0.01,0.006)), (1990,(0.93,0.3,0.0012)), (?,?)}

The last dimension of the new wolf is determined like the other dimensions.

Appendix 3. Fig. A-1.
The flowchart of the proposed method