A Service Clustering Approach Based on Semantics Ant Colony Algorithm

Zhou Xiang-bing[1,3], Luo qi[2]
[1] Dep of Software Key Lab of Sichuan Normal University,
Chengdu Sichuan 610068, China
8017zk@gmail.com

[2]School of computer science, Wuhan Institute of Technology
Wuhan 430073,China
luoqi@gmail.com

[3]Department of Mathematics , Aba Teachers College,
Chengdu Sichuan 611741, China

***Abstract*—In order to solve accuracy of Topic Maps of navigation and positioning for service clutering in the distributed software system,we proposed a navigation and positioning model of Topic Maps based on ontology relation(NPTMO), and according to NPTMO, we build a service clutering model and semantics ant colony algorithm. Therefore, which implement navigation and positioning of resources from relation of Topic Maps. Firstly, we define ontology relation and calculate approach, and analyze associate characteristic of Topic Maps based on ontology; Secondly, we analyze and study to match and merge approach of Topic Maps, and employ tag relation of RDF, OWL-S, WSDL and XTM what to build navigation and positioning system,and build a service clutering model, on the basic of these context above description. We design a semantisc ant colony algorithm(SACA) to calculate optimum solution of the model, and compared SACA with basic ant colony algorithm(ACA) showed that results of SACA was better than ACA. Finally, we define recall rate of ontology, accuracy of Topic Maps and efficiency of SACA to evaluate the approach showed that increased accuracy of Topic Maps connect resources.**

***Index Terms*—Ontology relation;Service clutering model; Navigation and positioning; Semantisc ant colony algorithm; Semantics web service composition**

I. Introduction

Topic Maps have been published as ISO 13250 standard since 1999, which efficiently apply many fields such as Enterprise Information Integration， web sites and portals， knowledge management and knowlege organization, e-learning, and more[1]. In fact, Topic Map is also a ontology, and has semantics recognition capability, which is a structuring semantics in the information resources layer, and employ relation to describe semantics among concept.Then obtain some different resources to guide other resources connection. Topic Map is different from ontolgoy, which provide a knowledge code topic-oriented and standard-based, and the semantics has been well defined before, which own the simultaneously description langage and query langage. To build a knowledge navigation interface. Therefore, is known as bridge of information management and knowledge management, which is a locator of GPS in the information resources. But Topic Map is important difference from Semantic Web, which is applied to description of concept formalization and classification. However, Topic Map is applied to implement formalization of index and glossary in the building process[2], therefore, give Semantics Web high accuracy degree. But interrelated research and application of Topic Maps is still developing. Therefore, many researcher attempt to contribute beneficially theory and approach, which has proposed a lot of approach and some open sources framework at present. For example, Jung-Mn Kim,Hyopil Shin and Hyoung-Joo Kim proposed schema and constraints-based matching and merging of Topic Maps[3], Myongho Yi proposed information organization and retrieval using a Topic Maps-based ontology-Results of a task-based evaluation[4], however open source software mainly have TM4J、TMTab、Omnigator、StarTree and more[5]. In particular, some international conferences on Topic Maps have been held, such as International Conference on Topic Maps Research and Applications(TMRA) have been held since 2005 at Leipzig, in Germany[6], which had discussed and proposed a lot of approach and application that was relevant to Topic Maps, included matching, similarity and description of Topic Maps, and other.

II. Relation Basic Knowlegde

Topic Map is composed of topic, association, occurrence, facet, Scope and other items; Which showed in Fig.1[11]. However, standard development of Topic Map is showed in Fig.2. At this time, characteristic and application of Topic Map can be summed up following ten aspects:

(1)Organize great informosome;
(2)Catch and organize web services;
(3)Stand for complex regulation and workflow;

doi:10.4304/jsw.4.9.1022-1031

(4)Support machine learning;

(5)Manage distributing knowlegde and information;

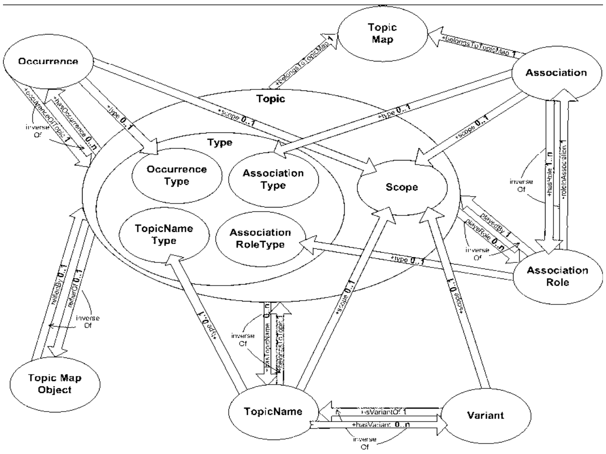
(6)Compose information and knowlegde;

(7)Have navigation and positioning capability of information resources;

(8)Have sematics discernment and extraction capability;

(9)Provide the best flexible structure;

(10)Possess facet classification of Topic Maps.



Fig,1 Architecture of Topic Maps

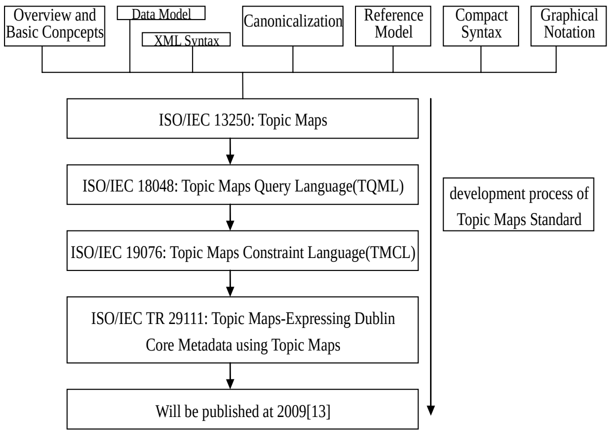


Fig.2 Development Process of Topic Maps Standard

Now, according to regulation and application of Topic Maps, we build a navigation and positioning framework of Topic Maps, which are simplified TMNLF; To implement serivce clustering model. Therefore, we divide TMNLF into five layers, which include application layer, sematics layer, information layer, resource layer and Topic Maps layer, showed in Fig.4. At the same time, we adopt semantics to implement resource extraction and recognition positioning, however, at present, structuring data storage employ relation database to store, such as SQLServer, Oracle,and so on. But semi-structured data have been applied many fields, especially web service application, and WSDL lack semantics description; which have a large challenge of semantics recognition and extraction in the web service, and for navigation and positioning of resource is also difficlut.therefore, we must solve four problems following:

(1)Between ontologly and RDF of mapping regulation;

(2)Between WSDL and RDF of mapping regulation;

(3)Among ontology, WSDL and RDF of navigation and positioning approach;

(4)Service clutering approach.

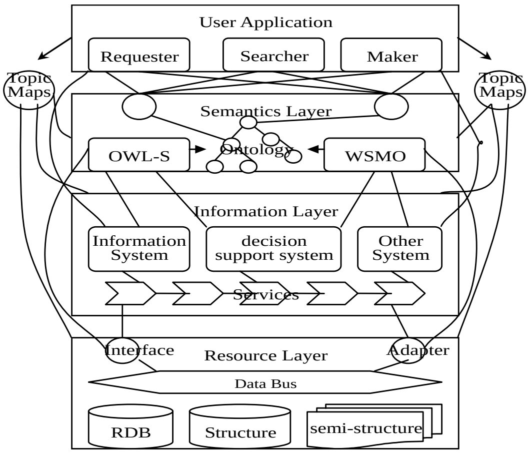


Fig.3 Layered Structure of Navigation and Positioning for Topic Maps

In the Fig.3, Application Layer: which is composed of service consumer, service provider,service maker, information searcher, services. To provide reslut of demand and feedback, and develop diffierence business logic fuction, and other.

Semantics Layer: which implement semantics interoperability and composition, generally employ OWL-S and WSDO to describe, based on ontology.

Informatiom Layer: all kinds of business systems compose the layer, and according to demand of application layer and recognition derivation of semantics layer meet business fuction of web service(WS) and semantics web service(SWS).

Resource Layer: all kinds of data resources compose the layer, those resources may be distributed difference fields, and the data storage is also varied, which is structured relation database, unstructured and semi-structured XML data, and other storage method.

Topic Maps layer: the layer is aim to navigation and positioning for application layer, semantics layer, resource layer and information layer.

III. Navigation and Positioning Model of Service Clustering Based on Ontology Relation

A. Theory Foundation of Navigation and Positioning Model

Definition 1 Ontology relation can be defined five tuples: $OR=<O,S,Dl,R,M(a,b)>$, where

(1)O is ontology, and $O=<C,P,R,H^C,rel,A^O>$[16], then a ontology set is $Os=\{os_1,os_2,\ldots,os_t\}$, which hierarchical concept $H^C(Arc)\subseteq C\times C$ is a directed relation; axiom set A^O must meet constraint of concept C and relation R in the ontology; Therefore, a ontology relation can define $Ont=<O, \xi,\psi>$, where, ξ is a ontology logic language, ψ is ontology structure dictionary. If $C=\{c_1,c_2,\ldots,c_q\}$ is O's concept node set, $R=\{r_1,r_2,\ldots,r_n\}$ is relation node set, then denote $Arc=(C\times R)\cup(R\times C)\cup(R\times R)\cup(C\times C)$.

(2)S denote source state of ontology, if define $S=\{s_1,s_2,\ldots,s_n\}$, then adopt $S_i(O_j)$ to denote S.

(3) DL denote description language of ontology, which include OWL-S and WSMO, and define $DL=<OS,WM,Sign>$[17].

(4)R is relation among ontology, for $\forall O\rightarrow S_i(O_j)$,$\exists DL\rightarrow f{:}O\Rightarrow R=(O\times O)$.

(5) $M(a,b)$ denotes mapping and matching approach of ontology.

1) To define ontology mapping: $M(a)=<So,To,Mr,P,I,Pf>$, which separately denote source ontology, target ontology, mapping relation, constraint parameter, mapping interface and support platform. Where,

$So=\{so_1,so_2,\ldots,so_n\}$, which employ tree structure $T_1(So,\alpha)$ to describe;

$To=\{to_1,to_2,\ldots,to_m\}$, which employ tree structure $T_2(So,\beta)$ to describe;

$P=\{(p_i,p_{i+1})\}$, which employ queue $Q(P)$ to express.

At the time, mapping relation denote: $Mr=((T_1(So)\times T_2(To)), Q(P))\cup((T_2(To)\times T_1(So)), Q(P))$.

If $Mr=\{mr_1,mr_2,\ldots,mr_s\}$ is a mapping relation set, and meet $Mr(P)\rightarrow<So,To>$; then adopt (1) to calculate similarity between source ontology and target ontology, and select the strongest similarity as mapping object.

$$Sim(So,To)=\frac{\sum\limits_{i,j\in Q(P);p\in So,q\in To;k\in Mr(P)}(|Os_{p,k}-T_1(So,\alpha)_i|)(|Os_{q,k}-T_2(To,\beta)_j|)}{\sqrt{\sum\limits_{i,j\in Q(P);p\in So,q\in To;k\in Mr(P)}(|Os_{p,k}-T_1(So,\alpha)_i|)^2+\sum\limits_{i,j\in Q(P);p\in So,q\in To;k\in Mr(P)}(|Os_{q,k}-T_2(To,\beta)_j|)^2}} \quad (1)$$

In the (1), which employ improved cosine theorem to define similarity, where, (p,k) and (q,k) separately denote corresponding relation of source ontology set and target ontology set to ontology set, and their corresponding relation probability separetely define $P(p\,|\,k)=\frac{|So|}{|Os|}$, $P(q\,|\,k)=\frac{|To|}{|Os|}$, and $So,To\subset Os$. k,i,j separetely denote parameters of mapping relation set, constraints and evaluate Ev of (p,k), (q,k). At the same time, define constraint degree $C_R(P,X)$ of $P\rightarrow X(So,To)$:

$$C_R(P,X)=\frac{\frac{|So|+|To|}{|Os|}\exp(Sim(So,To))}{2\exp(\sum\limits_{i=1}^{n}\sum\limits_{j=1}^{m}\sqrt{(P(p\,|\,k)_i-P(q\,|\,k)_j)^2})}$$

If $Ev\in[0,1]$, then define $Ev(x)=\int_0^1 Mr(P(x_{i,j}))dx$.

Algorithm 1 Ontology mapping algorithm.

Input:So,To

Output:The best result of ontology mapping

$\forall So,To\in Os,\exists M(a)\rightarrow OR$;

if(($So\cup To\rightarrow Mr$)&&($P\rightarrow(So,To)$))

$(So,To)\rightarrow M(a(I))[\alpha\xleftrightarrow{P}\beta]$; //Source ontology and target ontology shake hands via interface I in the $M(a)$,which use data structure with Tree and Queue to express.

for(int r=0;r<num;r++) //num is given number, and num$\in$n(natural number).

Evaluate(Given($E(x)$))$\rightarrow Rc$;

{ Computing($P(p|k),P(q|k)$)$\rightarrow V$ (V_1,V_2);// Obtain a pair of value.

if($\frac{|Os\cap To|}{|Os|}\leftrightarrow Value$)

Calculating and Running $\{Sim(So,To),C_R(P,X)\}\rightarrow V[Sim,C_R]$;// Obtain a group of Sim and C_R value.

Obtain(max($V[Sim,C_R],E(x)$))$\rightarrow$mapping(So,To);

return mapping result; }

2) To define ontology matching $M(b)=<Os,M(a)>$, if $\forall$ $O_i,O_j(\in Os)$'s membership function denote $M_{O_i}(a), M_{O_j}(a)$, then employ weight hamming formula to calculate distance between $M_{O_i}(a)$ and $M_{O_j}(a)$:

$R_w(O_i,O_j)=\frac{1}{n}\sum\limits_{i=1}^{n}w(a_i)\,|\,M_{O_i}(a_x)-M_{O_j}(a_x)\,|$, where

$W(a_i)$ is weight coefficient on the a_i, and meet $\frac{1}{n}\sum\limits_{x=1}^{n}w(a_x)=1$. And employ hamming neartude to calulate distance among ontology.

$N_H(O_i,O_j)=1-R_w(O_i,O_j)$

Now, we still define a ontology matching entropy, which reach accurately matching among ontology, thus empoly $O(M_O(a_1), M_O(a_2),\ldots, M_O(a_n))$ to denote ontolgoy vector, and we employ Draka entropy to express ontology matching entropy.

$$H(O)=\frac{1}{n\ln^2 n}\sum_{x=1}^{n}[-M_o(x_x)\ln M_o(a_x)-(1-M_o(a_x))\ln(1-M_o(a_x))]$$

According to above formula in the 2), we define matching formula following

$Match(O_i,O_j)=\kappa\frac{N_H(O_i,O_j)}{H(O)}\cdot R_w(O_i,O_j)$, where,

κ denote a proportionality coefficient between hamming neartude and Draka matching entropy.

Algorithm 2 ontology matching algorithm

Input: $M(b)$
Output: The best result of ontology matching
$\forall O_1, O_2 \in Os, \exists M_{O_1}(a), M_{O_2}(a) \rightarrow M(b)$;
for(int r=0;r<num;r++){
if($M(a)\rightarrow$ResultValue) //ontology mapping have succeeded.
$\forall (O_1, O_2 \in Os) \rightarrow O, \exists M_{O_i}(a), M_{O_j}(a) \rightarrow$ ResultValue;
Computing($R_w(O_i, O_j)$)$\rightarrow$$ValueR[M]$;
$N_H(O_i, O_j) = 1 - ValueR[M] \rightarrow ValueN[U]$;//reduce to calculate complexity.
Computing($H(O)$)$\rightarrow$$ValueH[N]$);
$\max\{Match(O_i, O_j) = \kappa \dfrac{ValueN[U]}{ValueN} Value[M]\}$;
return matching result; }

We employ a online bookshop system[18](OBS) to explain calculating approach of Algorithm 1 and Algorithm 2, and adopt OBS'query to explain, which include four query terms: Bookname,Author, Press, Full-text, is spimled to BAPF, showed in Fig.5, therefore, at the moment, $\exists$ontology set: $Os=\{os_1,...,os_{150}\}$, $\exists$source ontology set :$So=\{so_1,...,so_i(i\in[15,20])\}$ and target ontology set: $To=\{to_1,...,to_j(j\in[21,26])\}$, which calculated result showed in table 1 and table 2.

Table 1 Calculating approach of ontology mapping

So_i	To_j	$P(p\|k)$	$P(q\|k)$	$Sim(O_i,O_j)$	$C_R(P,X)$	$E(x)$(given)	Result of mapping(algorithm 1)
15	21	0.100	0.140	0.445	0.516	0.750	...$To_{23}\leftrightarrow So_{15}\leftrightarrow To_{25}\leftrightarrow So_{19}$...
16	22	0.107	0.147	0.482	0.502	0.810	... $To_{26}\leftrightarrow So_{16}\leftrightarrow To_{24}\leftrightarrow So_{18}$...
17	23	0.113	0.153	0.406	0.569	0.709	... $To_{25}\leftrightarrow So_{17}\leftrightarrow To_{22}\leftrightarrow So_{16}$...
18	24	0.120	0.160	0.505	0.511	0.855	... $To_{15}\leftrightarrow So_{18}\leftrightarrow To_{26}\leftrightarrow So_{20}$...
19	25	0.127	0.167	0.512	0.519	0.872	... $To_{18}\leftrightarrow So_{19}\leftrightarrow To_{21}\leftrightarrow So_{15}$...
20	26	0.133	0.173	0.457	0.567	0.800	... $To_{16}\leftrightarrow So_{20}\leftrightarrow To_{23}\leftrightarrow So_{17}$...

Table 2 Calculating approach of ontotogy matching

So_i	To_j	$R_w(O_i,O_j)$	$w(a_i)$ (given)	$N_H(O_i,O_j)$	$H(O)$	$Match(O_i,O_j)\kappa$=3/4	Result of matching(algorithm 2)
15	21	0.700	0.10	0.300	0.1642	0.959	...$To_{22}\infty So_{15}\infty To_{26}\infty So_{20}$...
16	22	0.008	0.05	0.992	0.1664	0.036	... $To_{23}\infty So_{16}\infty To_{25}\infty\ So_{16}$...
17	23	0.210	0.30	0.790	0.1621	0.768	... $To_{25}\infty So_{17}\infty To_{21}\infty\ So_{18}$...
18	24	1.575	0.35	0.000	0.1663	0.000	... $To_{26}\infty So_{18}\infty To_{23}\infty\ So_{17}$...
19	25	0.358	0.11	0.642	0.1660	1.000	... $To_{21}\infty So_{19}\infty To_{22}\infty\ So_{19}$...
20	26	0.450	0.09	0.550	0.1656	1.000	—

Definition 2 WS Ontology can be defined six tuples:

OWS=<*ID*,*DWSDL*,*WSS*,*UDDI*,*OR*,*QoS*>, where

ID is a ID set of WS Ontology; *DWSDL* is fuction description of WS Ontology; *WSS* is semantics set of WS Ontology; *UDDL* is registration set of WS Ontology; *OR* is relation to *WSS*, which denote description of ontology relation and SWS; *QoS* is NOT-function description of service composition, namely Quality of Service.

At this time, we define WS Ontology service composition: *OWSC*=<*OWS*,π>, where π denote a operation method of *OWSC*.

Definition 3[3] We define a Topic Map model as following seven tuples:

$TM=<T_C,T_O,T_A,T_R,T_I,R_H,R_A>$, where

T_C denotes a set of topic types; T_O denotes a set of occurrence types; T_A denotes a set of association types; T_R denotes a set of role types; T_I denotes a set of instance topics; R_H denotes a set of subsumption hierarchy relations; R_A denotes a set of associative relations.

And the same T_C play different T_R, namely $OR\rightarrow TM$.

Definition 4 Implementation of Topic Maps as following three tuples: *TAO*=<*TS*,*AR*,*RO*>, where

TS : adopt subject to express Topic; *AR*: adopt Relationship to express Association, *RO* denote connection of Occurrence and Topic.

And meet <*TM*,*OR*>$\rightarrow$*TAO*.

In connection with BAPF, we design six Topic Maps, and is distributed to three layer, which included Kownledge layer, Information layer and Business layer, showed in Fig.4.

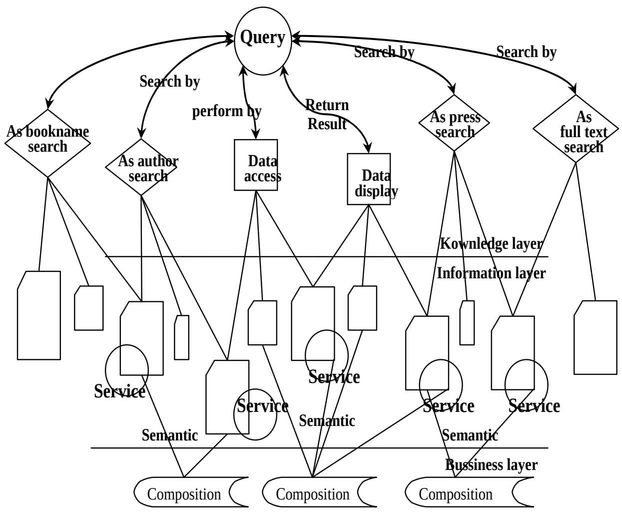


Fig.4 Query Topic Maps of OBS

Definition 5 Topic Maps matching define as following four tupls:

TMM=<*Re*,*TSim*(*TC*,*RH*,*RA*),*D*,*F*>, where

(1)*Re* denotes Topic Maps recognition, namely implement proccess of *OWS.OR* ;

(2)*TSim*(*TC*,*RH*,*RA*) denote a calculating approach of Topic Maps types, hierarchical relationship and incidence relation:

$$Sim_{TC}(TM_i, TM_j) = [(Dist_{TM,D}(TM_i, D) + Dist_{TM,D}(TM_j, D)) + (Dist_{Name}(TM_i, TM_j))]/3 \cdot TMCount$$

Which $Dist_{TM,D}$, $Dist_{Name}$ separately denote different distance of ontology and Topic.

If hierarchical structure of Topic Maps employ a height *h*'s *m* tree structure to description, then root of sub-tree adopt number *a* to denote, parent adopt number *b* to denote. Thus we define similarity of hierarchical relationship for TM_i and TM_j:

$$Sim_{RH}(TM_i, TM_j) = \left(\frac{Dis(TM_i.Child, TM_j.Child)}{(|a_i| + |a_j|)} + \frac{Dis(TM_i.Parent, TM_j.Parent)}{(|b_i| + |b_j|)}\right)/2$$

where

Dis() denote distance of child and parent of TM_i and TM_j.

And define similarity of incidence relation for TM_i and TM_j:

$$Sim_{RA}(TM_i, TM_j) = \frac{|TM_i| \cap |TM_j|}{|TM_i| \cup |TM_j|}$$

At the time, we define similarity of Topic Maps:

$$TSim(TM_i, TM_j) = \sqrt[3]{Sim_{TC} \cdot Sim_{RH} \cdot Sim_{RA}}$$

(3)*D* denotes query of Topic Maps, namely:query BAPF.

(4)*F* denotes entry point of matching facet of Topic Maps, such as different user search book. And distance among Topic Maps, we employ Euclid formula to calculate, thus, if Topic matching is attached to *OR*'s matching, and define membership function of TM_i and TM_j is $\mu_{TM_i}(x_i)$ and $\mu_{TM_{ji}}(x_j)$, then

$$Di(TM_i, TM_j) = \sqrt{\frac{1}{n}\sum_{i,j=1}^{n}[\mu_{TM_i}(x_i) - \mu_{TM_{ji}}(x_j)]^2}$$

Theorem 1 A web service ontology(WSO) is regarded as a Subject of Topic.

Description: WSO is a state of semantics service oriented ontology, services are direct correspondence *OWS.DWSDL*, and Topic Map is a group of incidence relation which connect resource throught Topics. In the oriented-service of software system, these resources can come from difference service fuction, however, in the paper, services are regarded as Topics. Thus

$\forall SWS \in OWS$, $\exists OWS.DWSDL \rightarrow SWS$;

$\forall TM$, $\exists$Subject identifier:*Sif* and subject indicator:*Sit*$\rightarrow$Subject;

Roof:

$\because \forall$OWS, $TM \rightarrow OR$;

$\therefore OWS.DWSDL \rightarrow OR, TM.T_C \rightarrow OR$;

$\therefore TM \rightarrow OWS.DWSDL$;

$\because TM\ (Sif, Sit) \rightarrow$Subject;

$\therefore$ Subject$\rightarrow OWS.DWSDL$;

Again

$\because OWS.DWSDL \in OWS$;

$\therefore OWS \rightarrow$ Subject.

Algorithm 3 Matching Algorithm of Topic Maps

Input: $TM \rightarrow TAO$;

Output: The best result of Topic Maps matching

$\forall(TM_i, TM_j \rightarrow TAO) \wedge (TM_i, TM_j \rightarrow OR)$, $\exists <(TM_i, TM_j), OR> \rightarrow TAO$;

if($\exists(TM.T_C, OR) \rightarrow TAO.TS$)

for(int *i*=1;*i*<num;*i*++) // num is given number, and num$\in$n(natural number)

{Find(*TMM.D* and *TMM.F*) // Find *D* and *F* of Topic Maps;

$VF = [\log_2 |TM.T_c / TMCount|] \mapsto TMM.F$:$Di(TM_i, TM_j)$// adopt *VF* as instruction of *TMM.F* selecting; *TMCount* is Topic Maps amount at the moment.

Computing $Di(TM_i, TM_j)$;

max($TSim(TM_i, TM_j)$);

}

return matching result;

Table 3 Matching calculating approach of Topic Maps(showed in Fig.4, different Topic assign difference ID number, showed in Table3. When users search, make Topic Maps count is 20, namely: *TMCount*=20)

$TM_{i(\mathrm{ID})}$	$TM_{j(\mathrm{ID})}$	$Di(TM_i,TM_j)$:VF	$Sim_{TC}(TM_i,TM_j)$ $\lvert D\rvert=4$	$Sim_{RH}(TM_i,TM_j)$ $a_i=12,a_j=15$ $b_i=18,b_j=20$	$Sim_{RA}(TM_i,TM_j)$	$TSim\ (TM_i,TM_j)$	algorithm 3
5	32	0.9546: 0.1161	0.9333	0.6272	0.7297	0.753108	$TM_{i(11)}\leftrightarrow \mathrm{TM}_{j(32)}$
11	26	0.0191: 0.1730	0.7333	0.4389	0.4054	0.507197	$TM_{i(9)}\leftrightarrow \mathrm{TM}_{j(30)}$
14	18	0.4950: 0.1904	0.4667	0.1926	0.1250	0.223976	$TM_{i(20)}\leftrightarrow \mathrm{TM}_{j(26)}$
9	7	0.0000: 0.1585	0.1667	0.5704	0.1250	0.228214	$TM_{i(14)}\leftrightarrow \mathrm{TM}_{j(10)}$
20	10	0.5000: 0.2161	0.5333	0.2198	0.3333	0.339322	$TM_{i(17)}\leftrightarrow \mathrm{TM}_{j(7)}$
17	30	0.65000:0.2044	0.8667	0.5537	0.2766	0.510112	$TM_{i(5)}\leftrightarrow \mathrm{TM}_{j(18)}$

In the Table 3's data, according to Fig.4, and we have developed six query Topic Maps, which implement in the open source software TM4J.

Definition 6 Topic Maps merge:

Merge:

$\forall OWS\rightarrow OWSC$;

$\exists TM\rightarrow TAO\Leftrightarrow(TM_A((OWS\times OWS:OR)\times TM_B(OWS\times OWS:OR))\wedge((TM_A\leftrightarrow TMM)\times (TM_B\leftrightarrow TMM))\rightarrow TM):TAO$.

Theorem 2 Semantics service composition direct influence in Topic Maps merge(obviously).

B. Approach of Service Clutering Description

At present, ontology description language have OWL-S and WSMO(Web Service Modeling Ontology), and we employ OWL-S to describe service clutering in the paper, This is because online bookshop system adopt OWL-S to describe. When implement ontology reasoning, which employ TransitiveProperty, SymmetricProperty, inverseOf, equivalent and other property to connect resource, namely: connect RDF, Finally accomplish ontology reasoning recognition; However, RDF(s) is a schema of description resource property, which assign property to connect resource. And the property is also resource, have property; showed in Fig.5.WSDL describe business fuction, but lack semantics description, therefore employ a special semantics language to improve semantics recognition capability of web services, which effectually implement service clutering, accormplish service composition.

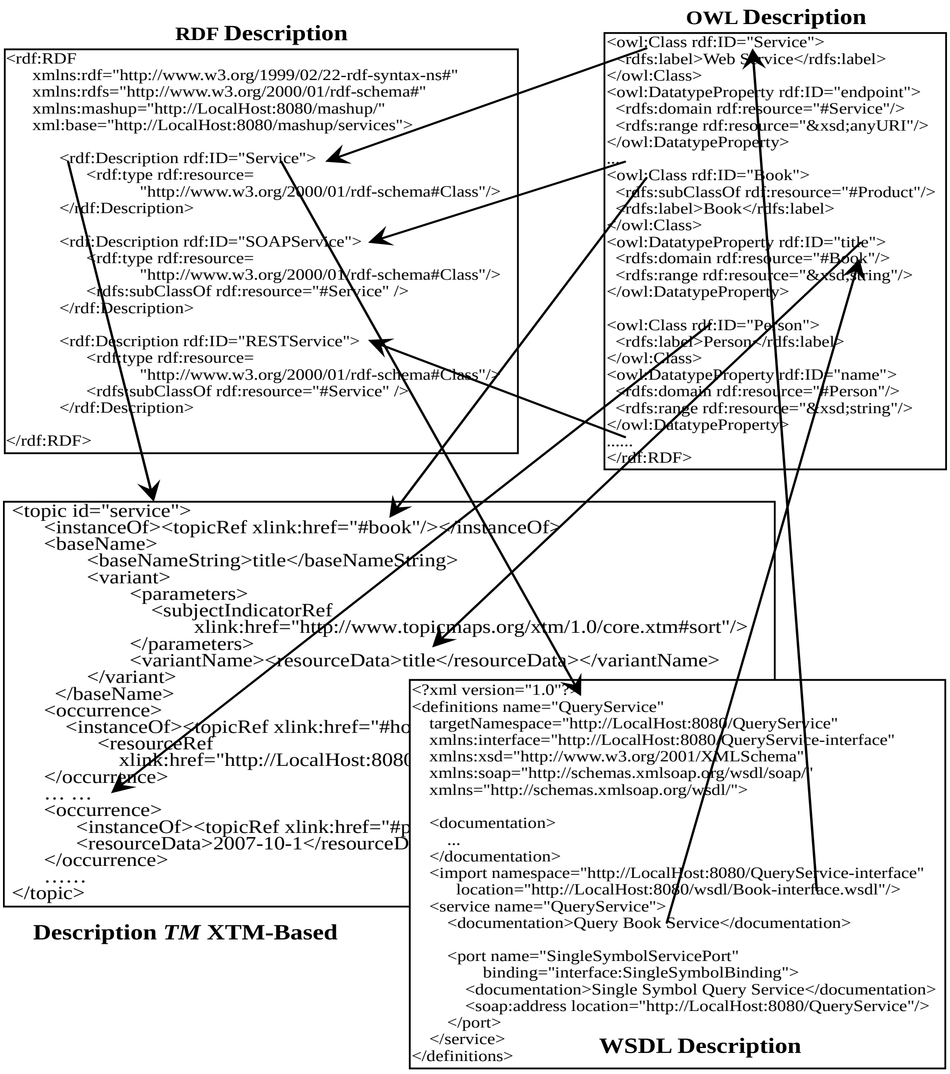


Fig.5 Service Clutering Description Approach

OWL-S[19] is semantics description language of web service based on OWL, which is built services concept, is composed of Profiles、Models and Groundings, and we join Topic Maps in OWL-S, showed in Fig.6 and Code List L1. Employ URI, parameter and OWL in mapping web service, showed in Fig.7. And L1 describe Topic Maps based on OWL-S, in order to adopt Topics to map web services, and implement business function.

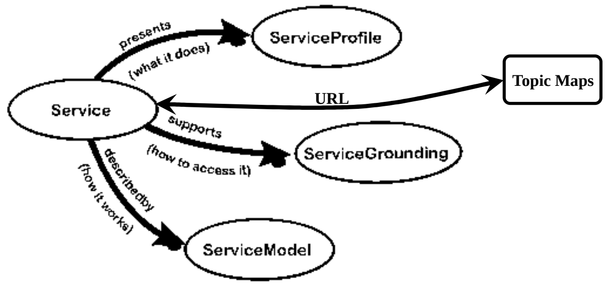


Fig.6 Join Topic Maps in Service Ontology

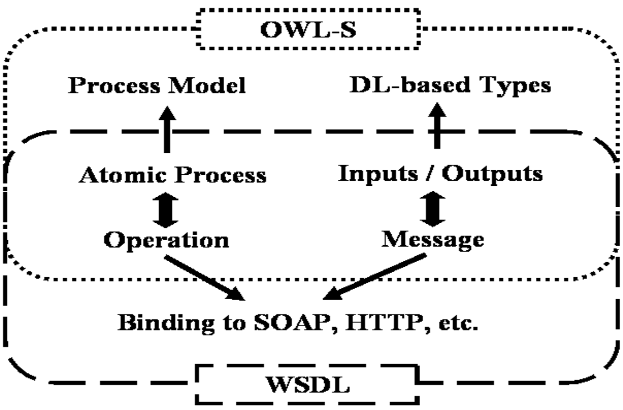


Fig.7 Mapping between OWL-S and WSDL

L1

```
<owl:Class rdf:ID="Topic Maps index">
<rdfs:subClassOf>
<owl:Restriction>
<owl:onResourceProperty rdf:resource="#XTM"/>
<owl:cardinality
rdf:datatype="&xsd;nonNegativeInteger">
1</owl:cardinality>
</owl:Restriction>
</rdfs:subClassOf>
<rdfs:subClassOf>
<owl:Restriction>
<owl:onResourceProperty rdf:resource="#Topic"/>
<owl:cardinality
rdf:datatype="&xsd;nonNegativeInteger">
1</owl:cardinality>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>
```

IV. SERVICE CLUTERING MODEL AND SEMANTICS ANT COLONY ALGORITHM

When implement service clutering model, we employ RDF Description, OWL Description, WSDL Description and Topic Maps Description based on XML, and adopt their tag relation to recognize sermantics and connect resource.Therefore:

Definition 7 RDF Description employ set $RdfS=\{<Rf_1,F_1>,<Rf1,F_1>,\ldots, <Rf_{n1},F_{n1}>\}$ to describe, which $<Rf_{n1},F_{n1}>$ denote RDF tag and resource description.

OWL-S Description employ set $OwlS=\{<Ol_1,O_1>,<Ol_2,O_2>,\ldots,<Ol_{n2},O_{n2}>\}$ to describe, which $<Ol_{n2},O_{n2}>$ denote OWL-S tag and resource semantics description.

WSDL Description employ set $WslS=\{<Wl_1,BF_1>,<Wl_2,BF_2>,\ldots,<Wl_{n3},BF_{n3}>\}$ to describe, which $<Wl_{n3},BF_{n3}>$ denote WSDL tag and business fuction description.

Topic Maps Description employ set $XtmS=\{<Xt_1,NP_1>,<Xt_2, NP_2>,\ldots,<Xt_{n4}, NP_{n4}>\}$ to describe, which $<Xt_{n4}, NP_{n4}>$ denote XTM tag and resource description.

Accroding to definition 7, we build service clutering model following:

$Ml=<RdfS, OwlS, WslS, XtmS>$

And tag set adopt Ml_{tag} to denote, and meet:

$(RdfS\times OwlS)\wedge(RdfS\times WslS)\wedge(RdfS\times XtmS)\wedge(OwlS\times WslS)\wedge(OwlS\times XtmS)\wedge(WslS\times XtmS)$

The research, we proposed a semantics ant coloy algorithm:SACA to optimize and implement service clutering model on basis of ant coloy algorithm, which operate Ml, therefore, we set four ant colonies, every colony have same or difference ants, then according to ant a hangover from the pheromone for a optimum path, namely: the path meet resources of service clutering, showed in Fig.8. And, according to similarity of tag string in the four fields, we build (*) model.

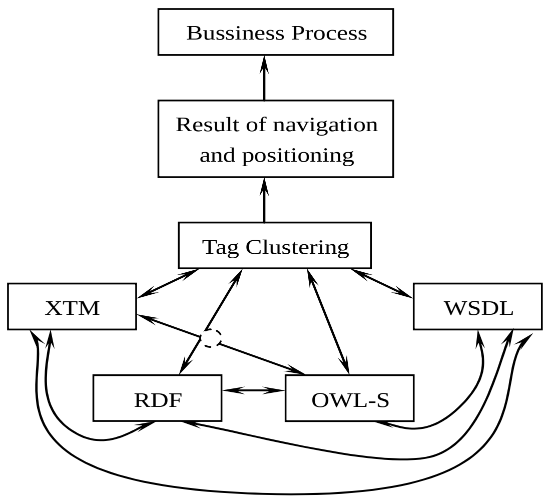


Fig.8 Service Clutering Model

$$f(String_i, x) = \max\{\sum_{String_j \in domain(x)} [1-\frac{1-Sim_{tag}(String_i, String_j)}{\alpha(1+(\upsilon-1)/\upsilon_{\max})}]\} \quad (*)$$

Where

x=1,2,3,4, which separately denote a set in the Ml; $String_i$,$String_j$ denote tag string in the Ml, Sim_{tag} denote tag's similarity in the Ml, $domain(x)$ express difference tag from different field, α denote similarity parameters, υ and υ_{max} separately denote ant velocity and maximum velocity.

When meet

Y=max[$Sim(So,To)$,$Match(O_i,O_j)$,$TSim(TM_i,TM_j)$], ∃

$Compare(f(String_i,1))\rightarrow\lambda_1$;

$Compare(f(String_i,2))\rightarrow\lambda_2$;

$Compare(f(String_i,3))\rightarrow\lambda_3$;

$Compare(f(String_i,4))\rightarrow\lambda_4$.

Which λ_i(i=1,2,3,4) is incidence matrix.

And set values following:

$$Compare(f(String_i,x)) = \begin{cases} 0(\text{disaffinity}) \\ 1/2(\text{similarity}) \\ 1(\text{difference}) \end{cases}$$

If $Sim_{tag}(String_i,String_j)$ employ KMP algorithm to calculate string matching

$$next[y] = \begin{cases} 0(y=1) \\ \max\{k \mid (1<k<y) \wedge \\ (p_1 \cdots p_{k+1} = p_{y-k+1} \cdots p_{y-1})\}(k \neq \Phi) \\ 1(\text{Other}) \end{cases}$$

Then $Sim_{tag}(String_i, String_j) = \dfrac{next[y]}{|p|}$, where $|p|$ denote $String_i$ length.

Algorithm 4 KMP algorithm

Input: Ml

Output: $Sim_{tag}(String_i,String_j)$

```
∃Some tags→ML;
if(Sim(So,To)&&Match(Oi,Oj)&&TSim(TMi,TMj))
  {for(int r=1;r<Z;r++)// Z'Value calculate method following
    {int Index_KMP(Sstring S, Sstring T, int pos, int CalS)
     {if((T≠Φ)&&(0≤pos≤StrLength(S)))
       i=pos;j=1;
       while(i<=S[0] && j<=T[0]){
         if(j==0 || S[i]==T[j]){++i;++j;}
         else j=next[j];}
      if(j>T[0]) return i=T[0];
      else return(0);
      Cals=StrLength(S);}}
int get_next(SString T, int &next[])
{i=1; next[1]=0;j=0;
while(i<T[0]){
  if(j==0 || T[i]==T[j]){ ++i;++j;next[i]==j;}
  else j=next[j];}}}
```

If probability of string $String_i$ tag r for $String_j$ matching is 1/4P_{rj}, then not-matching probability is (1-1/4 P_{rj}), $String_i$ for $String_j$ not-matching expectation is

$$E = \frac{1}{4}P_{ij} \cdot \prod_{r=1}^{Z}(1-\frac{1}{4}P_{rj})^{X_{rj}},$$
$$Z = |RdfS_{tag}| \, or \, |OwlS_{tag}| \, or \, |WslS_{tag}| \, or \, |XtmS_{tag}| \quad (2)$$

When implement tag clutering, firstly design program ς of navigation and positioning of Topic Maps oriented ontology relation, in order to build service clutering model, which not-matching expectation is minimum value, is build following (3) and (4), thereby improve tag matching efficiency of Ml.

$$E(\varsigma) = \min(\sum_{s=1}^{4}\sum_{i=1}^{A}\sum_{j}^{B}\{[\frac{1}{4}P_{ij}(\prod_{r=1}^{Z}(1-\frac{1}{4}P_{rj})^{X_{rj}})]w_s\}) \quad (3)$$

$$\begin{cases} A,B = |RdfS| \, or \, |OwlS| \, or \, |WslS| \, or \, |XtmS| \\ Z = |RdfS_{tag}| \, or \, |OwlS_{tag}| \, or \, |WslS_{tag}| \, or \, |XtmS_{tag}| \end{cases}$$

$$P_{ij} = 1 - f(String_i, x_i) \quad (4)$$

s.t.

$$(\text{i}) \sum_{j=1}^{A} X_{rj} = 1, r = 1,2,\cdots,Z$$

$$(\text{ii}) \sum^{Z} X_{rj} = 1 or 2 or 3 or 4$$

When implement ant colony algorithm to optimize, assume ants are m, and assign differently ants $m_i(k)$(m_1+…+m_4=m) to set in the Ml, then assign m_i to difference tag. If transition probability of ant is (5) and (6)[21], namely: probability of selection resource j for tag r.

$$Pp_k(r,j) = \begin{cases} \arg\max_{u \in allowed_k}\{\tau(r,u)\cdot\eta^{\beta}(r,u)\} & if \ q \le q_0 \\ J & else \end{cases} \quad (5)$$

$$J = \begin{cases} \dfrac{\tau(r,j)^{\alpha}\cdot\eta^{\beta}(r,j)}{\sum_{u \in allowed_k}\tau(r,j_u)^{\alpha}\cdot\eta^{\beta}(r,j_u)} & if \ j \in allowed_k \\ 0 & else \end{cases} \quad (6)$$

And build semantics pheromone of ($r\rightarrow j$) following (7) for navigation and positioning of Topic Maps.

$$\eta(r,j) = \sqrt[3]{\begin{matrix}\max[Sim(So,To)\cdot Match(O_i,O_j) \\ \cdot TSim(TM_i,TM_j)]\end{matrix}} \quad (7)$$

At this time, if ants k have assigned tag to build r in the $E(\zeta)$, then adopt $tabu_k$ to record tag that didn't match again, and according to constraint condition (i),(ii) adjust $E(\zeta)$, and adopt $allowed_k$={ML_{tag}-$tabu_k$} to express tag set assigned.

1) When ants k solve practicable solving $E(\zeta)$, according to (8) formula implement adjustment of local pheromone updated:

$$\tau(r,j) = (1-\rho)\cdot\tau(r,j)\cdot Sim_{tag}(String_i, String_j) + \rho\tau_0, 0 \le \rho \le 1 \quad (8)$$

Where

$\tau_0 = 1/(m \cdot E(\varsigma))$

2) According to (9) formula implement adjustment of global pheromone updated:

$$\tau(r,j) = (1-\theta)\cdot\tau(r,j)\cdot Compare(f(String_i,x)) + \frac{\theta}{E(\varsigma_{opvalue})}, 0 \le \theta \le 1 \quad (9)$$

Where

$E(\varsigma_{opvalue})$ denote current optimum expectation value.

Algorithm 5 Semantics ant colony algorithm

Step1 The initialization of tag scope every description language;

Step2 To calculate $f(String_i, x)$, and calculate max[$Sim(So,To)$, $Match(O_i,O_j)$, $TSim(TM_i,TM_j)$], which separately obtain the best data of ontology relation similarity, matching and the best data of Topic Maps similarity.

Step3 Employ KMP algorithm to implement tag matching every field, and obtain a group of effective matching value, and adopt (2) to calculate expectation value of not-matching tag.

Step4 Adopt (3) to calculate tag matching program of waiting assignment, and obtain P_{ij} in the (4).

Step5 According to constraint condition evaluate (2) and (3), and obtain matching program to assgin ants.

Step6 Adopt (5) to calculate $Pp_k(r, j)$, which implement ants to assgin, and accomplish difference navigation and positioning of tag, at the same time, adopt (6) to update optimun path of ants.

Step7 Adopt (7) to accomplish pheromone of ants optimun path to update. And (8) is a key of semantics service clutering.

Step8 What evaluate $E(\zeta)$ again, if don't meet demand of online bookshop system, then according to Step9 and Step10 implement local and global pheromone updated.

Step9 Adopt (8) to implement local pheromone updated.

Step10 Adopt (9) to implement global pheromone updated.

Step11 If meet demand, then end, else return Step2.

When implement semantics ant colony algorithm, we set m=50(m_1=12,m_2=15,m_3=8,m_4=15), q_0=0.7, α=1.5, β=2, ρ=0.2, θ=0.05 to experiment in the online bookshop system, which compared SACA with ACA, and adopt 20 times to experiment, iterative operation 2000 times, showed Fig.9. Thereby, SACA reach optimum solving at 600 times, and ACA reach optimum solving at 1600 times.

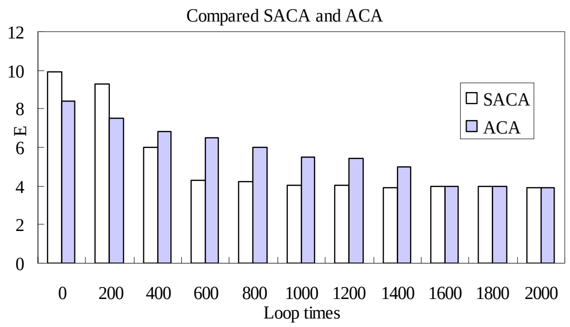


Fig.9 Result compared SACA with ACA

V. Experiment Result

Experiment results mainly make experiments from recall rate of web service ontology oriented ontology relation for navigation and positioning of Topic Maps, precision of Topic Maps, optimun calculation efficiency of SACA, and evaluate feasibility and advantage. Therefore, the experiment platfrom employ J2EE, and other open softwares which protégé[22]: ontology maker and knowledge acquisition, Axis[23]: development web service, TM4J[24]: development Topic Maps, saxon[25]: proccess XML, and so on. And we develop three program packages based on JAVA, which includes ontology relation package: Com.OntRelaTopicMapic.OntologyRelatin.*, navigation and positioning of Topic Maps package: Com. OntRelaTopicMapic.TopicMapicNP.*, semantics ant colony optimun package: Com.OntRelaTopicMapic.SACA.*. Finally, common integration make experiment in the Eclipse. Experiment system adopt online bookshop system, and experiment data make 30 services, 20 ontologies and 6 Topics. At the same time, set four counter, which separately count ontology(include source ontology and target ontology), Topic Maps, services(include resquest services and response services), service composition, thus define:

Recall rate of ontology relation:

$$Rcall = \frac{|So \cap To|}{|So \cup To|};$$

Precision of Topic Maps:

$$Pecis = \frac{|OWS \to Subject|}{|TM|};$$

Optimun efficiency of SACA:

$$Effici = \frac{|String_i \cap String_j|}{|Ml|}.$$

Experiment results showed in Table 4, Fig.10 and Fig.11.

Table 4 Adopt BAPF query to make experiments

Search item	*Recall*	*Pecis*	*Effici*
Bookname	95.653	96.843	94.500
Author	97.013	95.433	96.022
Press	94.788	95.578	94.778
Full-text	94.001	95.333	95.022

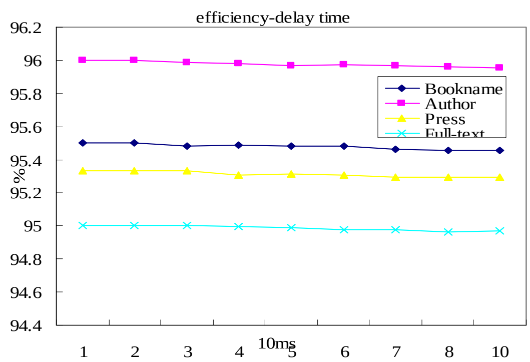


Fig.11 efficiency charge of BAPF along with delay time charge

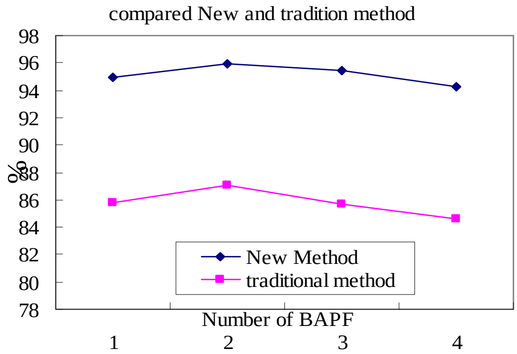


Fig.12 Efficiency compare new method with traditional method

VI. CONCLUSION

In the paper, we research condition of navigation and positioning of Topic Maps for service clutering model, therefore, firstly build software hiberarchy of navigation and positioning. Secondly, research ontology relation and describe language relation among RDF, OWL-S, WSDL and XTM; on basic of which build service clutering model, and build a semantics ant colony algorithm to optimize. In connection with these problems we define computational method, and give concrete algorithm and experiment data. Finally, employ online bookshop system to make experiment platform based on J2EE, and define recall rate of ontology relation, precision of navigation and positioning of Topic Maps and optimun efficiency of SACA. But alone with ontology and Topic Maps apply in all fields, and software system based on service is maturing,in this times, if improve the approach, efficaciously increase precision of service composition, thereby enhance user's demand. Especially we proposed semantics ant colony algorithm, which would improve semantics recognition and extraction capacity of service composition.

ACKNOWLEDGMENT

The Natural Sciences Foundation of Sichuan Provincial Education Bureau,China(No.07ZC002).

REFERENCES

[1] Oh, Sam G. MARC, FRBR and RDA: Topic Maps Perspective; Topic Maps 2008,http://www.topicmaps.com/tm2008/oh.ppt
[2] Jack Park and Sam Hunting. XML Topic Maps: Create and using Topic Maps for Web. Addison-Wesley,2002
[3] Jung-Mn Kim,Hyopil Shin and Hyoung-Joo Kim.Schema and constraints-based matching and merging of Topic Maps.Information Processing and Management,2007(43),pp.930–945
[4] Myongho Yi.Information Organization and Retrieval Using a Topic Maps-Based Ontology: Results of a Task-Based Evaluation.JOURNAL OF THE AMERICAN SOCIETY FOR INFORMATION SCIENCE AND TECHNOLOGY, 2008,59(12),pp.1898–1911
[5] Darina Dicheva and Christo Dichev.TM4L: Creating and browsing educational topic maps.British Journal of Educational Technology,2006,37(3),pp.391-404
[6] http://www.informatik.uni-trier.de/~ley/db/conf/tmra
[7] Librelotto, Giovani Rubert."Comparing topic maps constraint specification languages,"in Proceedings of Scaling Topic Maps-Third International Conference on Topic Maps Research and Applications(TMRA 2007), 2008,pp.86-97.
[8] Barta, Robert."Knowledge-oriented middleware using topic maps,"in Proceedings of Scaling Topic Maps -Third International Conference on Topic Maps Research and Applications(TMRA2007),2008,pp.98-115
[9] Kim, Jungmin."Topic maps matching computation based on composite matchers,".in proceedings of ICCS 2007 - 7th International Conference, 2007,pp. 696-703
[10] Motohro Mase,"Seiji Yamada and Katsumi Nitta.Extracting Topic Maps from Web Pages by Web Link Structure and Content," in proceedings of CEC2008,pp.1232-1239
[11] Anne Cregan."An OWL DL construction for the ISO Topic Map Data Model,"in proceedings of NICTA2005
[12] ISO/IEC 13250: Information technology – Topic Maps; International Organization for Standardization: Geneva, Switzerland, 2007
[13] ISO/IEC DTR 29111: 2007: Information technology – Topic Maps – Expressing Dublin Core Metadata using Topic Maps; International Organization for Standardization: Geneva, Switzerland, 2007,http://www.jtc1sc34.org/repository/0884.htm
[14] DU Xiao-Yong, LI Man1, WANG Shan.A Survey on Ontology Learning Research.Journal of Software,2006,17(9):1837-1847
[15] Gerardo Canfora, Massimiliano Di Penta, Raffaele Esposito,etl.A framework for QoS-aware binding and re-binding of composite web services.The Journal of Systems and Software 2008(81),pp.1754-1769
[16] Alexander Maedche. Ontology learn ing for the semantic web.Kluwer Academic Publishers: Boston /Dordrecht/ London. 2002.11-2l.
[17] ZHOU Xiang-bing, YANG Xing-jiang, XIANG Chang-cheng.SOA Component Localizer Mechanism and Algorithm Based on Ontology,Systems Engineering and Electronics, 2008,30(12):11-17
[18] http://www.ibm.com/developerworks/cn/offers/lp/ultimashup
[19] http://www.w3.org/Submission/OWL-S/
[20] A. Kaveh, B. Hassani, S.Shojaee, S.M.Tavakkoli.Structural topology optimization using ant colony methodology,Engineering Structures,2008(30),pp.2559–2565
[21] S.K. Chaharsooghi, Amir H. Meimand Kermani.An effective ant colony optimization algorithm (ACO) for multi-objective resource allocation problem (MORAP).Applied Mathematics and Computation ,2008(200),pp.167–177
[22] http://protege.stanford.edu
[23] http://ws.apache.org/axis/index.html
[24] http://www.tm4j.org
[25] http://saxon.sourceforge.net.