

# 基于边映射表连接的匹配算法\*

曹春萍, 何佳<sup>†</sup>

(上海理工大学光电信息与计算机工程学院, 上海 200093)

**摘要:** 针对面向语义网络图匹配的特殊性, 在基于状态回溯搜索算法的基础上提出一种新的称为基于边映射表连接的匹配算法, 利用语义网络图的有向性, 将图匹配问题转换为对搜索路径的规划, 并采用深度优先算法形成搜索步, 同时对目标图的所有边建立索引, 加快以边匹配为中心形成边映射表的过程, 最后对边映射表进行连接形成结果集。在真实数据集上的实验结果表明, 该算法具有较高的执行效率。

**关键词:** 语义网络; 图匹配; 边映射表

**中图分类号:** TP391      **文献标志码:** A      **文章编号:** 1001-3695(2013)09-2697-03

**doi:**10.3969/j.issn.1001-3695.2013.09.034

## Graph matching algorithm based on edge map's connection

CAO Chun-ping, HE Jia<sup>†</sup>

(School of Optical-Electrical & Computer Engineering, University of Shanghai for Science & Technology, Shanghai 200093, China)

**Abstract:** Aiming at the semantic network graph matching problem's particularity, this paper proposed a new matching algorithm based on the state backtracking search method, which called EBMG. This method took use of the semantic network's directional, converting the graph matching problem into a path search planning and used the depth-first algorithm to form a search step set. And at the same time this paper created index for the all edges of the target graph to accelerate the process of forming edge mapping table, finally made connect of the edge mapping table to get the result set. Experiment results on the real dataset show that the algorithm has higher efficiency.

**Key words:** semantic network; graph matching; edge mapping table

语义网络是一种知识表达形式, 它通过简单的语义三元组构成的有向标记图数据结构提供了强大而灵活的知识表示能力, 可方便地被机器所识别和应用, 是目前知识表示方式研究的热点之一。图路径搜索、图匹配和图变换是语义网络上的三大核心知识机制, 语义网络上的识别、推理、搜索以及语义网络上的过程计算等都需要应用图匹配和图变换计算。针对图匹配问题, 许多研究者提出了一些非常有效的算法, 比较经典的有 Subdue 算法<sup>[1]</sup>、Ullman 算法<sup>[2]</sup>、Nauty 算法<sup>[3]</sup>, 以及后人在这些算法基础之上改进的 Graph Explore 算法<sup>[4]</sup>、Quick-SI 算法<sup>[5]</sup>、VF2 算法<sup>[6]</sup>和基于信息熵的子图匹配算法<sup>[7]</sup>等, 这些算法本质上都是经典回溯搜索算法或索引预处理算法的变种。但是语义网络上的图匹配过程有其特殊性, 首先, 语义网络是节点和边带标记的有向图, 语义网络上的图匹配是基于图元素标记的匹配, 而标记匹配的标准由不同的应用环境决定, 因此难以进行索引的预处理, 并且用户输入的模式图是灵活多变的, 修改索引模型的代价太大; 其次, 语义网络的目标图往往是整个以语义网络表示的知识集, 规模巨大, 回溯搜索会消耗太多的资源, 因回溯算法一般采用递归函数调用, 在运行时如果调用层次太深会出现堆栈溢出的问题; 最后, 语义网络图匹配中往往存在歧途的问题, 即在匹配过程中往往会在目标图中找到大量的无效的部分匹配的子图, 而这些子图的无效是因为较早期的状态的节点没有满足匹配条件, 回溯算法在这一点没有

进行处理, 从而影响匹配的效率。

本文针对经典回溯搜索算法在进行语义网络图匹配中存在的缺陷进行了改进, 提出一种新的称为基于边映射表连接的匹配算法, 利用语义网络图的有向性, 将图匹配问题转换为对搜索路径的规划并采用深度优先算法形成搜索步, 同时对目标图的所有边建立索引, 加快以边匹配为中心形成边映射表的过程, 最后对边映射表进行连接形成结果集。该算法有效地实现了语义网络上快速的图匹配过程。

### 1 基于边映射表连接的匹配算法

图匹配是以一个图作为模式图、以另一个图作为目标图、在目标图上找到与模式图匹配的图或子图的过程。为了阐述问题的需要, 首先给出语义网络的定义以及图匹配的形式化定义。

**定义 1** 语义网络。一个语义网络通常是一个有向图, 有向图  $G$  定义为一个四元组  $(V, E, I, L)$ ,  $V(G)$ 、 $E(G)$  分别是顶点集和边集,  $E(G)$  中的每个元素都是由  $V(G)$  中两个不同元素组成的集合,  $L$  为标记集,  $I$  为  $V \cup E$  到  $L$  的映射, 称为标记映射, 分别有边标记  $I(e)$  和节点标记  $I(v)$ 。

**定义 2** 图匹配。一个图匹配  $M$  是从一个模式图  $G^P$  到目标图  $G^R$  的映射  $M^G(G^P, G^R): G^P \rightarrow G^R$ ,  $M^G = \langle m^V, m^E \rangle$  是对于映射  $f$  的一对映射, 其中模式图和目标图都是属于图的概念,  $m^V$  是节点映射  $m^V: v^P \xrightarrow{f} v^R$ ,  $m^E$  是边映射  $m^E: e^P \xrightarrow{f} e^R$ , 模式

**收稿日期:** 2013-01-07; **修回日期:** 2013-03-01      **基金项目:** 国家自然科学基金资助项目(61074087); 上海市教育委员会科研创新基金资助项目(12ZZ144)

**作者简介:** 曹春萍(1968-), 女, 甘肃兰州人, 副教授, 硕士, 主要研究方向为图像处理、智能数据处理(ccpged@163.com); 何佳(1987-), 女(通信作者), 硕士, 主要研究方向为语义计算。

图中的节点和边称为模式节点和模式边,目标图中的节点和边称为目标节点和目标边。

基于边映射表连接的匹配算法 (edge-based graph matching) 参考了基于状态回溯搜索的匹配算法, 仍然将图匹配问题作为一个累进的部分子图匹配问题, 首先将模式图拆分为一系列递增的子图, 由递增的子图规划出由多个搜索步组成的搜索路径, 每个搜索步包含一条边和边连接的两个节点信息, 通过其包含的信息去从目标图中匹配出满足标准的多个匹配状态, 搜索步对应的匹配状态形成一个个映射表。语义网络是一个有向图, 利用节点和节点的有向性, 可以从特定的映射表出发, 按照一定的规则进行映射表的连接, 得到最终的结果子图。本文提出的算法具体有以下步骤: a) 递增子图和搜索步的构造; b) 针对每个搜索步, 以边匹配为中心, 获取搜索步对应的映射表; c) 对映射表作连接, 获得最终的结果子图。

### 1.1 递增子图和搜索步的构造

将模式图拆分为一系列的递增子图, 然后以这些子图为依据递增地找到目标图上递增的子图是基于状态回溯搜索的匹配算法的基本思想。图 1 展示的是对模式图拆分为递增子图的过程。

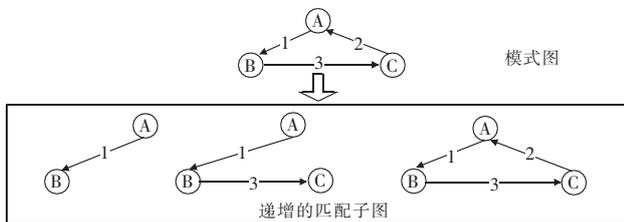


图 1 模式图拆分成递增子图

如图 1 所示, 递增的子图可以转换为一条搜索路径, 一条搜索路径由多个搜索步组成, 对图 1 而言, 相应的搜索路径为 A1B→B3C→C2A。递增子图和搜索步的构造其实质是针对一个模式图, 从某个节点出发进行深度优先遍历的过程<sup>[8]</sup>, 同时返回一条搜索路径。在该过程中需要存储每个搜索步, 搜索步由一条边以及连接的两个节点组成, 如 A1B。该过程可以如下算法描述。

#### 算法 1 基于深度优先的搜索步构造算法

输入: 模式图  $G^p$ , 起始节点  $v$ 。

输出: 搜索路径  $L = \{l_1, l_2, \dots, l_n\}$ 。

- 从起始节点  $v$  出发, 标记为已访问过。
- 依次从  $v$  出发搜索每个与  $v$  连接的边  $l(v, w, t)$ , 如果  $w$  没有访问过, 则组成一个三元组  $\langle v, l, w \rangle$  纳入  $L$  集合中, 并以  $w$  为新的出发点, 循环进行步骤 b), 直到图中所有和  $v$  相连的边均已被访问为止。
- 若图中仍有未访问的节点, 则选取另一个尚未访问的顶点作为新的起始节点, 转步骤 b), 直到图中所有节点均已被访问为止。
- 返回  $L$ 。

end

### 1.2 映射表的建立

假设有一个如图 2 所示的简单的目标图, 对每一个搜索步, 去目标图中匹配出相应的子图, 比如对搜索步 A1B, 先在目标图中搜索到边 1, 然后对边 1 所连接的节点进行节点匹配, 匹配出来的结果称之为映射子图, 一个搜索步可能会匹配出多个映射子图, 每一个搜索步得到的映射子图组成一个映射表。对每个子图进行边的扩展, 直到映射子图的穷尽, 最后的结果就是匹配结果, 这个过程其实相当于对映射表的连接操作, 如下所示。

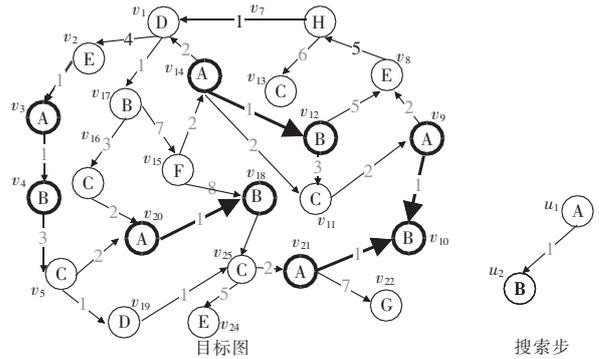


图 2 以边为中心的匹配过程

例如, 对搜索步 A1B 而言, 以边 1 为搜索起点, 从目标图中可以搜索到节点对  $(v_7, v_1)$ 、 $(v_9, v_{10})$ 、 $(v_1, v_{17})$ 、 $(v_3, v_4)$ 、 $(v_{14}, v_{12})$ 、 $(v_{20}, v_{18})$ 、 $(v_{21}, v_{10})$ 、 $(v_{19}, v_{25})$ 、 $(v_5, v_{19})$ , 再对边 1 连接节点的起点和终点进行匹配, 符合条件的将有  $(v_3, 1, v_4)$ 、 $(v_{20}, 1, v_{18})$ 、 $(v_{14}, 1, v_{12})$ 、 $(v_9, 1, v_{10})$ 、 $(v_{21}, 1, v_{10})$ , 如图 2 中粗线标注部分所示。经过上述过程, 由搜索步 A1B 得到的映射表如表 1 所示。对搜索步 B3C 而言, 搜索边 3 如图 3 所示, 从目标图中可以搜索到节点对  $(v_4, v_5)$ 、 $(v_{17}, v_{16})$ 、 $(v_{18}, v_{25})$ 、 $(v_{12}, v_{11})$ , 再对边 3 连接节点的起点和终点进行匹配, 符合条件的将有  $(v_4, 3, v_5)$ 、 $(v_{17}, 3, v_{16})$ 、 $(v_{18}, 3, v_{25})$ 、 $(v_{12}, 3, v_{11})$ , 因此由搜索步 B3C 得到的映射表如表 2 所示。

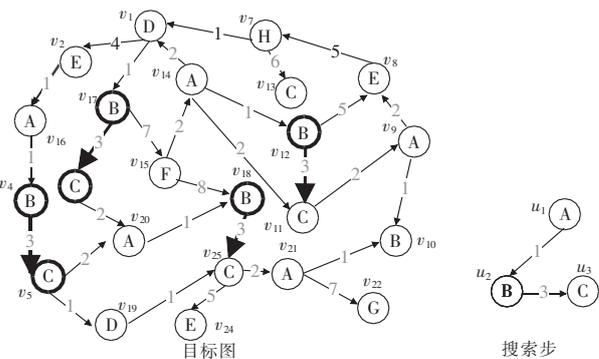


图 3 以边为中心的匹配过程

表 1 搜索步 A1B 的映射表

$u_1$	$u_2$
$v_3$	$v_4$
$v_9$	$v_{10}$
$v_{14}$	$v_{12}$
$v_{20}$	$v_{18}$
$v_{21}$	$v_{10}$

表 2 搜索步 B3C 的映射表

$u_2$	$u_3$
$v_4$	$v_5$
$v_{17}$	$v_{16}$
$v_{18}$	$v_{25}$
$v_{12}$	$v_{11}$

对映射表 1 和 2, 依据边 1 指向的节点  $u_2$  作连接, 会得到如下节点集:  $(v_3, 1, v_4, 3, v_5)$ 、 $(v_{20}, 1, v_{18}, 3, v_{25})$ 、 $(v_{14}, 1, v_{12}, 3, v_{11})$ 。对比目标图, 由搜索路径 A1B→B3C 得到映射表连接出来的子图是匹配的。事实上, 对于模式图  $G^p$  和目标图  $G^R$ , 对  $G^p$  所有的边, 作以边为中心到  $G^R$  的映射, 对所得到的映射表作连接, 将所得的结果中包含重复节点信息的行删除, 就是  $G^p$  到  $G^R$  的匹配结果。

由上述的过程可知, 在对每个搜索步进行匹配时, 是以边为匹配中心的, 为了快速完成对边的搜索匹配过程, 可以首先对目标图的边建立如下的索引<sup>[11]</sup>:

$$L - \text{index}(l) = \langle L_R(l), \langle l_i, v_{in}, v_{out} \rangle \rangle, \text{其中 } L_R(l) \text{ 代表边集}$$

合,包含边标记信息和位置信息, $\langle l_i, v_{\lambda}, v_{\mu} \rangle$ 表示边  $l_i$  连接的入节点和出节点通过  $L-index(l)$  可以迅速定位到特定的边及其所连接的节点,而无须遍历整个目标图来搜索对应的边信息,在运行的过程中大大提高了边映射表形成的速度。边映射表连接举例如图 4 所示。

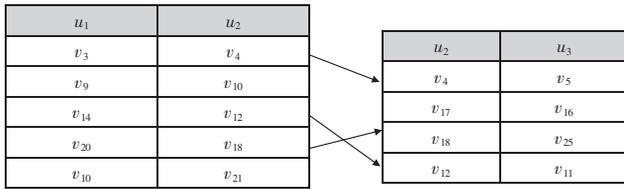


图 4 边映射表连接举例

### 1.3 基于映射表的连接算法

对建立了边索引结构的目标图而言,匹配算法效率能否提高的关键在于映射表连接效率的高低。事实上,经过了 1.2 节“映射表的建立”这一步骤,已经得到了模式图中关于所有边的边映射表,如果存在  $l_i \in E(G^p)$  的边映射表为空,说明该条边以及连接的节点在目标图中没有匹配的子图,则模式图必定与目标图是不匹配的,边映射表的连接算法可以不用进行便能返回结果。以下讨论的是  $\forall l_i \in E(G^p)$ ,其边映射表都不为空的情况。

假设模式图  $G^p$  经过搜索路径规划后,拥有了  $n$  个搜索步,即对应地具有了  $n$  个边映射表,按照搜索步给边映射表编号,分别为  $\{m_1, m_2, m_3, \dots, m_n\}$ ,对这  $n$  个边映射表进行连接通常有两种方式:第一种称为广度优先连接,是从  $m_1$  开始,先将  $m_{i-1}$  和  $m_i$  进行连接,然后得到全部结果之后再与  $m_{i+1}$  进行连接,直到  $m_n$ ;第二种方式是从  $m_1$  开始,先将  $m_{i-1}$  和  $m_i$  进行连接,得到一条结果后,再与  $m_{i+1}$  连接,直到  $m_n$ 。如果在这一步中连接某张表时发现无法进一步进行扩展,那么就返回到上一张表;如果一条结果经过所有映射表的扩展,则说明由该记录扩展出来的最后结果是一个跟目标图匹配的子图,该方法称为深度优先连接。这两种方法在理论上都是有效的,对第一种方法而言,当边映射表的记录数目总和非常大的情况下,内存无法全部存下,需要临时存储一部分在磁盘上,频繁地、无规律地访问各个数据碎片,会产生大量的数据 IO 操作,从而会极大地降低算法的效率。对第二种方法,只要每次产生了一条有效的结果,就立即进行下一条边的扩展,所以对于边投影表数据访问不够集中,可能某一条当前有效的结果在下一步立马失效,又得回溯到上一步,因此导致 IO 操作的数量也十分庞大,没有充分利用现有资源的大小。

事实上,对于本文面向语义网络的图匹配过程而言,语义网络必定是一个有向图,边总是有向的,这样就确定了边映射表连接的方向性。只要保持映射表连接的方向与边方向是一致的,在映射表内部之间的连接顺序是可以调整的,即可以先进行  $m_{i-1}$  和  $m_i$  的连接,对产生的结果集  $m_{i-1,i}$  再与  $m_{i-2}$  或者  $m_{i+1}$  连接,对最终的结果是没有影响的。而对含有记录数目较小的映射表作连接,产生有效的结果集在一般概率上更小,本文便利用这一特性,首先对记录数最少的相邻表作连接,同时为了满足对计算实时性的需求,在该过程中,根据内存的大小对中间结果的数目进行了调整,不是得到一个连接所有的结果或者一条结果再进行下一步扩展,中间结果的数目是动态改变的,即设置缓冲区的大小。当连接结果使得缓冲区满了之后,

再开始对该缓冲区内的中间结果进行下一步的扩展,这样便防止了高频率的 IO 操作,算法的效率也随即提高了。

### 算法 2 基于边映射表的连接算法

输入:边映射表集合  $E(m) = \{m_1, m_2, m_3, \dots, m_n\}$ 。

输出:匹配子图集合  $RS$ 。

- 对  $E(m)$  中每个元素的记录数作一次遍历,取得每个元素的记录数目,记为  $Count_i$ 。
- 创建  $n$  个大小为  $s$  的中间结果缓冲区,  $rc_1, rc_2, \dots, rc_n$ 。
- for  $t = 1$  to  $t = n - 1$ ,找到  $count_t + count_{t+1}$  最小的两个相邻元素,转 d)。
- 对  $m_t$  和  $m_{t+1}$ ,  $u$  和  $v$  分别是  $m_t$  和  $m_{t+1}$  的出节点和入节点。
- 如果  $m_t$  不为空,for  $k = 1$  to  $k = count_t$ ,从  $m_t$  中取出一条记录,同时删除  $m_t$  中的该条记录,写入  $rc_t$ ,直至  $rc_t/2$  满。
- 对  $rc_t$  中的每一个  $u, m_{t+1}$  中找寻到对应的  $v$  进行连接,写入  $rc_t$ ,直至  $rc_t$  满。
- 如果  $m_t$  中还有未处理的结果  $r$ ,更新  $E(m)$  中  $m_t$ 。转 c)。
- 对  $rc_1, rc_2, \dots, rc_n$  缓冲区中包含的记录进行连接,写入  $RS$ 。转 a),直到  $E(m)$  中记录为空。
- 返回  $RS$ 。
- end

## 2 EBGM 算法与经典图匹配算法的实验对比

本实验是基于图数据库 Neo4j 的存储环境,且匹配标准设定为“概念精确匹配”,即模式图与目标图对应的节点和边的标记必须完全一致,本文主要在匹配效率上进行算法的对比。本文使用了两个目标图,即基于数据集 UCL-credit 和数据集 Webdata 建立的。对每个目标图设计了三个模式图进行匹配  $\{G_1^p, G_2^p, G_3^p\}, \{G_4^p, G_5^p, G_6^p\}$ ,模式图中边的个数为 10 ~ 15 不等。其中模式图结构的定义如表 3、4 所示。实验选取 VF2、Ullman 两种比较经典的图匹配算法,它们的实验比较结果如图 5、6 所示。图表中的数据都是通过多次运行程序取平均值所得,其中 EBGM 算法所用时间包含边映射表的生成时间。

表 3 由 UCL-credit 构建的目标图以及模式图的结构

UCL-credit	$G_1^p$	$G_2^p$	$G_3^p$
边数 $E$	14000	10	13
节点数 $V$	14700	13	20
不同的节点标签数 $I(V)$	79	8	9
不同的边标签数 $I(E)$	86	9	11

表 4 由 Webdata 构建的目标图以及模式图的结构

UCL-credit	$G_4^p$	$G_5^p$	$G_6^p$
边数 $E$	10625	12	14
节点数 $V$	10238	16	23
不同的节点标签数 $I(V)$	206	12	20
不同的边标签数 $I(E)$	98	9	11

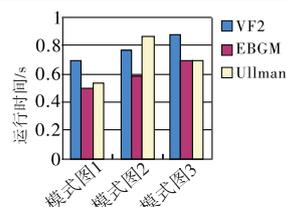


图 5 目标图  $G_1^p$  的三种算法时间比较

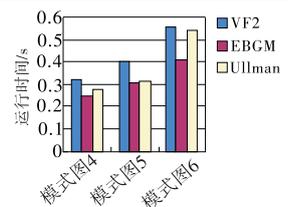


图 6 目标图  $G_2^p$  的三种算法时间比较

从图 5 和 6 中可以看出,EBGM 算法在不同复杂度的目标图 and 不同复杂度的模式图下都具有更高的时间效率。

## 3 结束语

本文针对语义网络上现存图匹配算法的缺陷,提出了以边为匹配中心的匹配算法,通过将模式图拆分成递增子图随后根据递增子图构造搜索步,将模式图对目标图的 (下转第 2703 页)

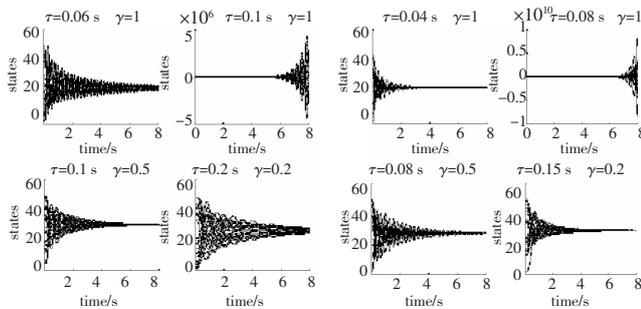


图7 无向规则拓扑下带延时的离散高阶算法的收敛性

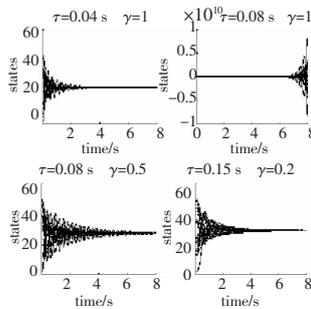


图8 无向随机拓扑下带延时的离散高阶算法的收敛性

## 4 结束语

本文提出了离散高阶分布式一致性算法,该算法通过利用二跳邻接节点前多步信息来加速一致性算法。当满足一定条件,离散高阶分布式一致性算法能收敛到初始状态的平均值。离散高阶分布式一致性算法在不增加通信量的前提下,其收敛速度大大高于相同条件下的伪二跳一致性算法收敛速度。当存在通信延时,离散高阶分布式一致性算法存在延时累计的现象,可以通过选择较小的参数 $\gamma$ 来提高其在通信延时情况下的稳定性。

### 参考文献:

- [1] OLFATI-SABER R, MURRAY R M. Consensus problems in networks of agents with switching topology and time-delays [J]. *IEEE Trans on Automatic Control*, 2004, 49(9): 1520-1533.
- [2] OLFATI-SABER R, FAX J A, MURRAY R M. Consensus and cooperation in networked multi-agent systems [J]. *Proceedings of the IEEE*, 2007, 95(1): 215-233.
- [3] ZHANG Ya, TIAN Yu-ping. Consensus of data-sampled multi-agent systems with random communication delay and packet loss [J]. *IEEE Trans on Automatic Control*, 2010, 55(4): 939-943.
- [4] BOYD S, GHOSH A, PRABHAKAR B, *et al.* Gossip algorithms: design, analysis and applications [C]//Proc of IEEE INFOCOM. 2005: 1653-1664.
- [5] MATHUR R, SHARMA M K, MISRA A, *et al.* Energy-efficient deployment of distributed mobile sensor networks using fuzzy logic systems [C]//Proc of International Conference on Advances in Computing, Control, & Telecommunication Technologies. 2009: 121-125.

(上接第2699页)匹配问题转换为基于边映射表的连接问题,并有效地提高了匹配过程的效率。但是该算法还存在以下一些需要改进的地方:a)本文提出的算法大大增加了存储需求,从搜索步的构造到映射表的建立,都需要存储大量的中间结果,因此需要构建更有效的存储结构;b)实验选择的匹配标准是“精确标记匹配”,但是对其他一些匹配标准,如关系匹配、层级概念序列匹配和相似度匹配等还没有提供实验证明。下一步工作是研究这些匹配标准对应的映射方法,然后提供利用该匹配算法的实验结果。

### 参考文献:

- [1] KETKAR N S, HOLDER L B, COOK D J. Subdue: compression-based frequent pattern discovery in graph data [C]//Proc of the 1st International Workshop on Open Source Data Mining. New York: ACM Press, 2005: 71-76.
- [2] CORDELLA L, FOGGIA P, SANSONE C, *et al.* A sub graph isomorphism algorithm for matching large graphs [J]. *IEEE Trans on Pattern Analysis and Machine Intelligence*, 2009, 26(10): 1367-1372.

- [6] AYSAL T C, ORESHKIN B N, COATES M J. Accelerated distributed average consensus via localized node state prediction [J]. *IEEE Trans on Signal Processing*, 2009, 57(4): 1563-1576.
- [7] FRANCESCHELLI M, GIUA A, SEATZU C. Distributed averaging in sensor networks based on broadcast gossip algorithms [J]. *IEEE Sensors Journal*, 2011, 11(3): 808-817.
- [8] RAJAGOPAL R, WAINWRIGHT M J. Network-based consensus averaging with general noisy channels [J]. *IEEE Trans on Signal Processing*, 2011, 59(1): 373-385.
- [9] THANOU D, KOKIOPOULOU E, PU Y, *et al.* Distributed average consensus with quantization refinement [J]. *IEEE Trans on Signal Processing*, 2013, 61(1): 194-205.
- [10] OLFATI-SABER R. Ultra-fast consensus in small-world networks [C]//Proc of American Control Conference. Pasadena, CA: IEEE Press, 2005: 2371-2378.
- [11] JIN Z P, MURRAY R M. Random consensus protocol in large-scale networks [C]//Proc of the 46th Conference on Decision and Control. New Orleans: IEEE Press, 2007: 4227-4232.
- [12] XIAO Lin, BOYD S. Fast linear iterations for distributed averaging [J]. *Systems & Control Letters*, 2004, 53(1): 65-78.
- [13] AYSAL T C, YILDIZ M E, SARWATE A D, *et al.* Broadcast gossip algorithms for consensus [J]. *IEEE Trans on Signal Processing*, 2009, 57(7): 2748-2761.
- [14] JIN Zhi-pu, MURRAY R M. Multi-hop relay protocols for fast consensus seeking [C]//Proc of the 45th IEEE Conference on Decision and Control. 2006: 1001-1006.
- [15] KIM Y, MESBAHI M. On maximizing the second smallest eigenvalue of a state-dependent graph laplacian [J]. *IEEE Trans on Automatic Control*, 2006, 51(1): 116-120.
- [16] REN Wei, ATKINS E. Distributed multi-vehicle coordinated control via local information exchange [J]. *International Journal of Robust and Nonlinear Control*, 2007, 17(10-11): 1002-1033.
- [17] 彭焕新, 戚国庆, 盛安冬. 伪多跳中继分布式一致性算法 [J]. *控制理论与应用*, 2012, 29(5): 623-628.
- [18] YANG W, ANDREA L, BERTOZZI I, *et al.* Stability of a second order consensus algorithm with time delay [C]//Proc of the 47th IEEE Conference on Decision and Control. 2008: 2926-2931.
- [19] MEYER C D. Matrix analysis and applied linear algebra [M]. Philadelphia: Society for Industrial and Applied Mathematics, 2001.

- [3] 代印唐. 基于语义网络的知识协作关键技术的研究 [D]. 上海: 复旦大学, 2009.
- [4] 黄云, 洪佳明, 覃遵跃. 大型网络中近似子图匹配研究 [J]. *计算机工程*, 2012, 38(18): 50-52.
- [5] 陶松桥, 黄正东. 基于属性邻接图匹配的装配图模型搜索方法 [J]. *计算机辅助设计与图形学学报*, 2011, 2(3): 25-27.
- [6] 解春欣, 汪卫. 子网同构验证算法 OES [J]. *计算机工程*, 2011, 37(3): 94-97.
- [7] 马帅, 曹洋, 沃天宇, 等. 社会网络与图匹配查询 [J]. *计算机学会通讯*, 2012, 8(4): 78-81.
- [8] 张治, 施鹏飞. 一种有效的贪婪模式匹配算法 [J]. *计算机研究与发展*, 2007, 44(11): 1903-1911.
- [9] BUNKE H. Graph matching: theoretical, foundations, algorithms and applications [J]. *IEEE Trans on Pami*, 2009, 3(9): 504-519.
- [10] 魏国辉, 杨春德, 谭军. DNA 计算机中图的深度优先搜索遍历算法 [J]. *计算机工程*, 2008, 34(15): 102-104.
- [11] 赵宏伟, 张海龙, 刘萍萍, 等. 基于表象式语义网络的图匹配算法 [J]. *吉林大学学报*, 2008, 11(S1): 35-37.