

# 基于 Bitmap 的油水井采注优化实时推理引擎

刘阳<sup>1</sup> 张天石<sup>1</sup> 李世超<sup>1</sup> 佟星<sup>1</sup> 曾鹏<sup>1</sup> 于海斌<sup>1</sup>

**摘要** 针对油田油水井采注优化业务中, 油水井数据量大、地层结构复杂以及人类经验多的特点, 分析了传统推理方法在油田采注实时优化处理过程中的不足, 采用事件处理思想, 提出了一种基于 Bitmap 事件编码与匹配机制的推理引擎, 有效地实现了对无效事件的过滤并提升了事件与规则的匹配效率. 在油田实际数据试验平台上对该方法进行了验证并与 RETE 算法、LFA (Linear forward-chaining) 算法的性能对比, 结果验证了本文方法在实时推理能力上的有效性.

**关键词** 采注优化, 推理引擎, Bitmap, 规则匹配, 事件过滤

**引用格式** 刘阳, 张天石, 李世超, 佟星, 曾鹏, 于海斌. 基于 Bitmap 的油水井采注优化实时推理引擎. 自动化学报, 2017, 43(6): 1007–1016

**DOI** 10.16383/j.aas.2017.c170132

## A Real-time Reasoning Engine for Injection-production Optimization of Water and Oil Wells on Account of Bitmap

LIU Yang<sup>1</sup> ZHANG Tian-Shi<sup>1</sup> LI Shi-Chao<sup>1</sup> TONG Xing<sup>1</sup> ZENG Peng<sup>1</sup> YU Hai-Bin<sup>1</sup>

**Abstract** In the process of optimizing the injection-production in oil and water wells, complex stratigraphic structure and large amount of business data and human experience will be involved. In this situation, traditional reasoning methods cannot be effective. By introducing the event processing theory, a reasoning engine using bitmap event encoding and matching is proposed, which can filter out invalid events efficiently and improve the matching performance between events and rules. The proposed reasoning engine is implemented in a real oilfield data experiment platform. Compared with RETE algorithm and LFA (Linear forward-chaining) algorithm, the proposed method shows a better reasoning capability.

**Key words** Production-injection optimization, reasoning engine, bitmap, rule matching, event filtering

**Citation** Liu Yang, Zhang Tian-Shi, Li Shi-Chao, Tong Xing, Zeng Peng, Yu Hai-Bin. A real-time reasoning engine for injection-production optimization of water and oil wells on account of bitmap. *Acta Automatica Sinica*, 2017, 43(6): 1007–1016

新一代信息技术的发展推动了物联网技术的产业推进. 在工业领域, 工业物联网已经成为工业泛在信息获取的重要依托手段, 改变了过去流程工业内部生产信息以及环境信息难以实时获取的问题, 成为工业领域全流程优化的重要使能手段.

近年来, 随着工业物联网体系在油田领域的部署与推广, 已经初步实现了数字油田的建设目标, 未来智慧油田将成为提升油田开发效率、降低成本、提升竞争力的重要发展方向. 对比数字油田侧重于基于泛在感知技术对油田各种生产和管理数据的采集、上传与存储, 智慧油田更加倾向于融入人类的经

验与智慧, 通过对采集到的各种勘探、开发、生产、运营及外部环境等数据进行全面整理和全局分析, 为指导高效、低成本、实时的开采措施制定与运营决策提供辅助支持. 在我国, 很大一部分油田开发进入中后期阶段, 面临含水率不断升高、能效比逐渐降低的问题. 对于油田油井、水井管网进行采注协同优化, 是提升油田作业效率的重要举措之一, 将是未来智慧油田的重要组成部分. 传统油田油水井采注调整通常采用“以采定注”的方式来进行, 缺乏充分利用油水井地层结构、数值模拟分析结果以及生产动态数据综合分析的工具手段, 在企业负责处理分析的不同业务部门在不同时间尺度分别进行优化策略制定的情况下, 综合采注优化周期较长. 随着工业物联网终端以及大数据平台的部署, 大量实时动态数据的产生, 这种由各部门在不同时间尺度上来联合驱动优化的模式将难以满足油田提升采注效率的需求, 融合油田工程师分析经验并基于油水井动态生产作业数据、油藏地质信息以及数值模拟信息的综合推理系统将有效改变油田企业多部门人员联合驱

收稿日期 2017-03-10 录用日期 2017-05-04  
Manuscript received March 10, 2017; accepted May 4, 2017  
国家自然科学基金 (61533015) 资助  
Supported by National Natural Science Foundation of China (61533015)

本文责任编辑 乔俊飞  
Recommended by Associate Editor QIAO Jun-Fei  
1. 中国科学院沈阳自动化研究所工业控制网络与系统研究室 沈阳 110016  
1. Lab of Industrial Control Network and System, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016

动优化的局面。

油田油水井数量庞大,会产生大量实时数据以及待处理分析事件,为适应对大量实时事件的快速处理,油水井采注协同优化推理引擎需要具有对大量同时到达的实时事件的实时处理能力,并能适应专业人员对推理引擎中的规则及参数进行修改。在融合人类经验方面,产生式推理引擎具有较好的用户适应性。产生式推理引擎可以根据用户定义的规则和当前产生的事件进行规则匹配并进行冲突消解,将产生的推理结果反馈给用户或者运行系统进一步操作使用的一种推理机制。如图 1 所示,一个产生式系统通常包括一组规则集,其中的每个规则都包括一组条件和执行动作,当当前外部事件与规则集中的规则条件在经过模式匹配后都得以满足时,该规则的执行动作将被加入到议程中,如果该动作是最终执行动作,那么该动作将被推送入执行推理结果引擎,如果该动作为推理中间结果,那么将被放入工作内存中等待后续事件到达继续执行。

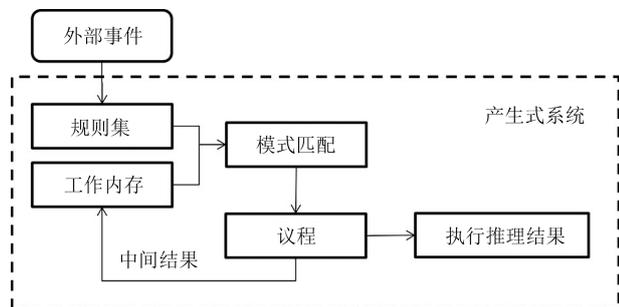


图 1 产生式推理引擎示意图

Fig. 1 The sketch map of production rule reasoning

可以看到,产生式推理引擎对人类经验融入的接口比较友好,但是其最大缺点是在大量规则集的情况下同时到达大量外部事件时,其匹配效率将十分低下,因为按照每个事件条件和规则逐一匹配,匹配的事件复杂度将与规则数量和事件数量的增长呈指数相关特点。

因此,本文以提升事件与规则的匹配效率为出发点,通过对事件和规则进行基于 Bitmap 的编码,提升了大数据事件与规则的匹配效率,并针对规则匹配过程中无效数据事件对推理引擎资源的占用问题,提出了对无效事件的分组与处理机制,实验结果表明了基于 Bitmap 编码的推理引擎的有效性。

## 1 相关工作

油田开发生产的优化过程控制中,会产生实时处理大量事件及规则的需求,为应对大数据量事件带来的处理问题,采用对事件的编码压缩与索引的方法可以有效提升对事件处理的效率,其中 Bitmap

是应用比较广泛的一种处理机制。针对大数据压缩以及索引问题, O'Neil<sup>[1-2]</sup> 最先提出 Bitmap 索引技术, Bitmap 就是用一个 bit 位来标记某个元素对应的数据,由于采用了 bit 为单位来存储数据,因此可以大大节省存储空间; Bitmap 技术中主要包括了数据的压缩、分类以及检索<sup>[3]</sup>。许多学者提出了不同的 Bitmap 压缩以及检索算法。Antoshenkov<sup>[4]</sup> 提出了 BBC (Byte-aligned bitmap compression) 算法用于支持快速的数据压缩,同时保持了竞争压缩率 (Competitive compression rate); Wu 等<sup>[5]</sup> 提出的 WAH (World-aligned-hybrid) 检索算法用于索引和搜索像科学模拟结果以及网络流量这样数据量较大的数据集; Deliège 等提出的 PLWAH (Position list word aligned hybrid) 基于 WAH 算法更好地利用压缩空间以及 CPU, 在同一个条件下能够减少一半的存储空间<sup>[6]</sup>; Fusco 等<sup>[7]</sup> 提出了一种利用单词码本来大量减少 Bitmap 索引尺寸的算法 COMPAX (Compressed adaptive index); Kim 等<sup>[8]</sup> 提出了一种新的 bitmap 压缩算法 SBH (Superbyte-aligned hybrid), 该算法不仅是当前最先进的 Bitmap 压缩方案, 而且其处理速度是 WAH 的 5 倍。清华大学文羽豪等提出了 Bitmap 索引压缩算法<sup>[9]</sup>、SECOMPAX (Compressed adaptive index) 以及 Bitmap 索引编码算法 MASC<sup>[10]</sup>。

产生式推理是一类基于知识的模式匹配方法,是实现知识自动化的重要手段,王飞跃<sup>[11]</sup> 提出知识自动化可以理解成为基于知识的服务、基于决策的服务,关键是把数据信息与任务无缝、准确、即时的匹配与结合起来; RETE 算法是规则匹配中的一类经典方法<sup>[12]</sup>; RETE 算法是由 Forgy<sup>[13]</sup> 在 1979 年首先提出并进行了论述,根据规则生成一个 RETE 规则网络<sup>[14]</sup>,事件流不断通过 RETE 规则网络进行匹配,一直持续到网络的终端节点才结束,同时产生复合事件<sup>[15]</sup>。RETE 适用场景十分广泛;其中 Kawakami 等提出了一种基于 RETE 算法分别用于网络拓扑节点管理<sup>[16-17]</sup> 以及楼宇节能管理<sup>[18-19]</sup>。Pallavi 等<sup>[20]</sup> 应用 RETE 算法提高专家系统的应用效率,使得系统返回的信息更令人满意。Wu 提出了线性正向推理算法 LFA (Linear forward-chaining)<sup>[21]</sup>,通过拆分规则因子和规则执行体来减少模式匹配的比对次数,进而提升产生式推理系统的性能。燕山大学冯建周等<sup>[22]</sup> 提出了利用语义本体技术建立关联数据的物联网语义关联网络,通过 Jena 事件推理实现语义推理从而实现物联网系统的智能决策。在现有油田开采过程中应用模式匹配用于生产还处于起步阶段,朱秀莉等<sup>[23]</sup> 提出基于 T-S 推理网络预测油田开发指标,对于预测问题具有一定潜力。

在上述研究中, 虽然研究人员针对 RETE 算法的性能做了诸多改进, 但是针对油田推理引擎这种具有到达事件量大、规则数量大且随时有修改需求的推理场景涉及较少, 因此, 本文针对油田油水井采注联合优化需求, 本文提出了一种基于 Bitmap 的实时推理引擎, 通过引入 Bitmap 的处理特点来提升事件与规则的匹配效率并过滤无效事件。

## 2 预备知识

### 2.1 油田实时推理系统

在图 2 中, 油田专业用户可以通过规则集可视化接口对推理系统中的规则进行增加、修改和删除等操作, 实时推理引擎可对当前所有规则集进行规则冲突检测, 当新规则与规则集中其他规则有冲突时, 推理引擎需要提醒用户该冲突以及冲突产生原因, 该功能可有效避免由于不同用户在经验上不一致, 导致的推理结果不准确问题; 当用户录入的新规则不存在冲突时, 该规则可被直接存储到规则库中。当外部事件到达时, 该规则可以立即成为推理引擎的一部分并被执行。此外, 在油田实时推理引擎中, 数以万计的油水井将实时产生大量与采注作业相关的生产事件到达推理引擎, 对于这些到达事件的处理需要更多的计算资源, 而且具有紧急响应需求的事件 (例如油水井工况异常) 需要实时处理, 本文中将通过到达事件处理模块, 实现对油田场景的处理。

### 2.2 Bitmap

Bitmap 就是采用比特位来标记某个元素在指定前提下对应的值或者存在情况的方法, 是一种常

用的数据库索引, 由于采用比特位为单位对数据进行存储, 因此可以节省存储空间, 并可通过对比特位的布尔运算提升检索效率。

假设规则  $R$  的条件部分 (Left hand side, LHS) 为  $R_{condition}$ 。

$$R_{condition} = \{ (key_1, operation_1, value_1), (key_2, operation_2, value_2), \dots, (key_n, operation_n, value_n) \}$$

其中  $(key_i, operation_i, value_i)$  表示  $R_{condition}$  中第  $i$  个条件,  $key_i$  表示第  $i$  个条件的关键字,  $operation_i$  表示第  $i$  个条件的操作符,  $value_i$  表示第  $i$  个条件的阈值。在 Bitmap 映射中, 将全部规则集中的条件无重复列出, 并按照关键字  $key$  的升序进行排列, 形成基于规则集条件的升序列表。假设一个基于规则集条件的升序列表为  $L(R_{condition\_all})$ 。

$$L(R_{condition\_all}) = \{ ("薄层累计注水倍", ">", "7"), ("单层含水值", ">", "97\%"), ("单层含水值", "<", "75\%"), ("厚层累计注水倍数", ">", "6") \}$$

$L(R_{condition\_all})$  中共包含 4 个元素, 那么对于事件“单层含水值 = 80%”来说, 其对应升序列表  $L(R_{condition\_all})$  的 Bitmap 形式为  $B(E_{单层含水值}) = (0110)$ , 其中若第  $i$  位为“1”则表示事件与  $R_{condition}$  中的第  $i$  个元素的关键字相关, 若为“0”则表示不

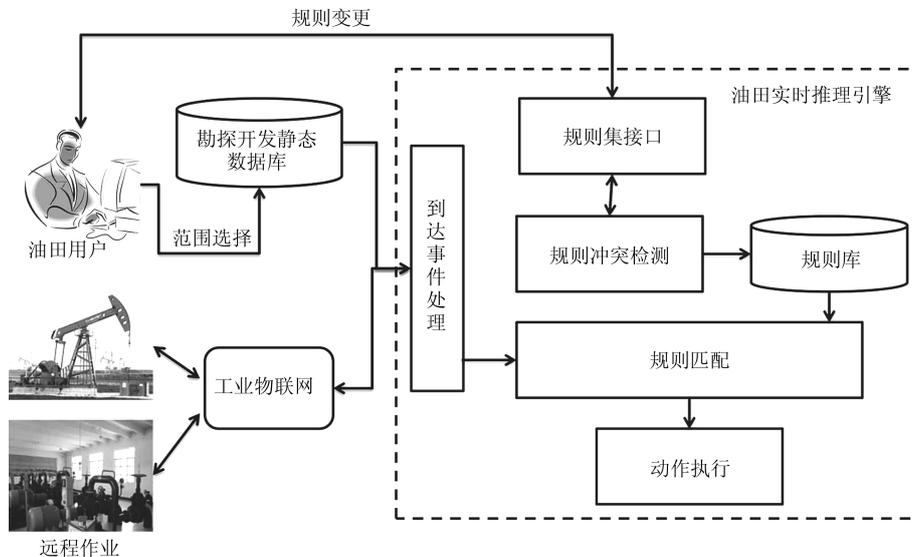


图 2 油田实时推理系统

Fig.2 Real-time reasoning system in oilfield

相关 (注: 本文中对于事件的 Bitmap 形式仅针对事件中的 *key* 部分来进行, 而不考虑事件中对事件取值的判断). 在事件“单层含水值 = 80%”中, 按照每个中文字符型占用 2 字节存储空间计算, 该事件的关键字“单层含水值”需要 10 字节存储, 而对应的 Bitmap 只需要 4 比特, 大大减少了事件的占用空间, 也提升了事件对于相关规则的定位效率.

此外, 基于  $L(R_{condition\_all})$  也可生成规则条件编码, 例如规则条件  $R_{condition} = \{(\text{“薄层累计注水倍数”}, >, \text{“7”}), (\text{“单层含水值”}, >, \text{“97%”})\}$ , 那么  $B(R_{condition}) = (0110)$ , 通过计算  $B(E_{\text{单层含水值}}) \wedge B(R_{condition}) = (0110) \neq B(R_{condition})$ , 可以得出当前事件不具备规则条件  $R_{condition}$  的全部执行条件, 不需要进行后续事件值与规则值的匹配判断.

可以看出, 事件关键字的表达与规则条件的升序列列表中的元素个数相同, 当规则条件升序列列表中元素个数较多, 例如为 10 000 时, 对应的事件关键字 Bitmap 长度也将达到 10 000 比特, 这将远远大于原始 10 字节的存储空间. 但是在采用规则条件升序列列表前提下, 事件的 Bitmap 中为“1”的比特位会集中出现, 其余位均为 0, 针对这种情况的 Bitmap, 其压缩方法已有很多, COMPAX 编码<sup>[7]</sup> 可以较好地解决该问题.

### 3 基于 Bitmap 的推理方法

图 3 所示为实时推理引擎架构, 主要包含到达事件处理和推理条件过滤两部分. 到达事件处理模块完成对达到数据的事件处理以及编码功能, 推理条件过滤模块完成对编码后事件的筛选与原子条件触发与推理执行功能.

#### 3.1 到达事件处理

在油田实时推理引擎中, 将主要接收两类报文, 一是由工业物联网传感器获取的实时生产及工况数据, 另一类是由油田用户选定的需要分析的静态勘探开发数据. 由于这两类报文来源不同, 导致其表现形式各异, 需要对这些数据进行预处理. 本文将这些具有复杂结构的周期作业报文以及油田勘探开发数据库中的各类数据采用事件封装的形式来统一处理, 对于实时推理引擎来说, 统称为到达事件. 需要注意的是, 这两类数据的位置标定方式略有不同, 其中传感器获取的数据通常以“井号, 时间”为中心进行描述存放在油田开发动态库中, 而油田静态勘探开发数据主要为油田工程师使用, 使用方式着眼于油水井与地层之间的关系, 通常以“井号 + 区块 + 层号”来使用相应地层地质信息, 因此需要对这两类数据报文进行基于井号的事件位置对齐. 为简化对异构数据源的处理, 本文给出如下定义:

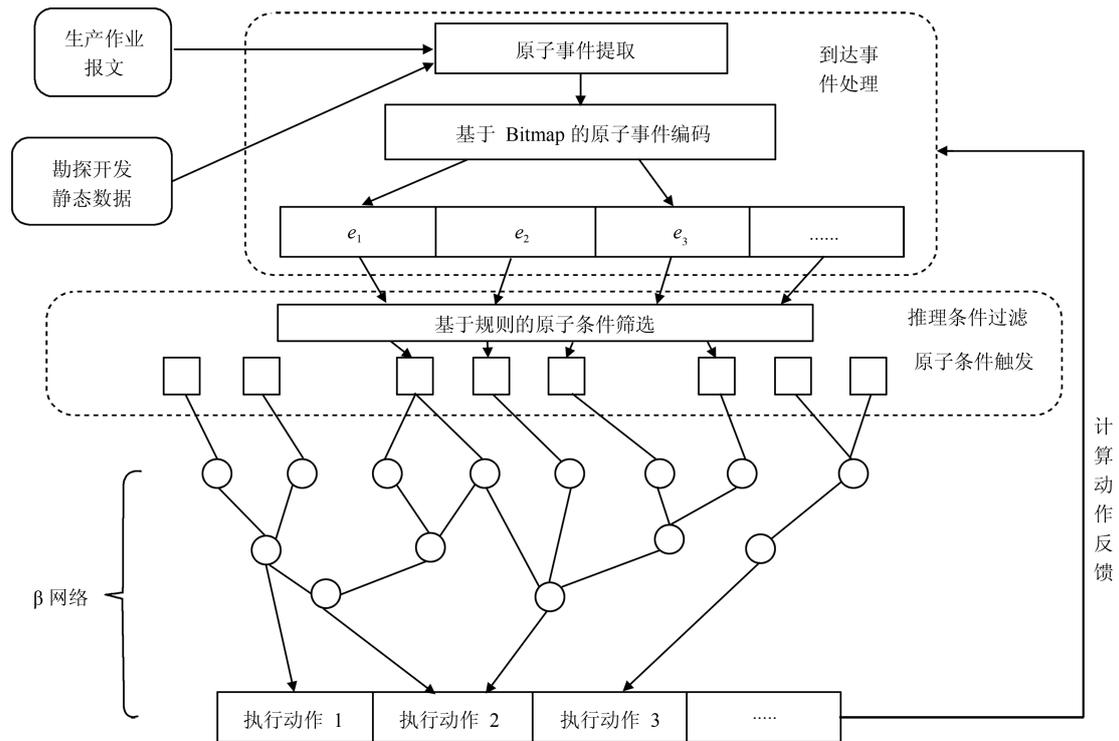


图 3 实时推理引擎架构

Fig. 3 The architecture of real-time reasoning engine

**定义 1.** 原子事件: 指仅包含一种数据类型的事件.

**定义 2.** 原子规则: 指规则中的 RHS (Right hand side) 部分仅能通过同时满足全部 LHS 条件才能触发的规则, 即原子规则中的 LHS 只能为单条件规则或者只包“ $\wedge$ ”运算符的条件组合.

**定义 3.** 原子条件: 原子规则中具有不可再分特征的条件.

例如, 对于具有单井温度、油压、套压、电参信息的复杂工业物联网传感器报文, 可以通过原子事件提取拆分成 4 个原子事件, 其中温度、油压、套压等信息称为原子事件的关键字, 原子事件表达形式为  $e_i = (k_i, v_i, geo_i, time\_window_i)$ , 其中  $k_i, v_i, geo_i, time\_window_i$  分别代表事件  $e_i$  的关键字、值、地理位置和时间窗口. 在形成原子事件后, 要完成对事件中关键字和地理位置的编码.

对关键字的编码采用基于 Bitmap 的编码方式, 首先对规则集中的所有原子条件进行基于条件关键字的升序排列, 形成原子条件升序列表  $L(AtomicCondition)$ , 然后对  $L(AtomicCondition)$  中的不同关键字起止位置构建 B+ 树  $Tree(AtomicCondition)$ . 对于到达事件中的关键字, 通过对  $Tree(AtomicCondition)$  的检索, 将对应起止位的比特值置为“1”, 其他位置为“0”, 然后采用 COMPAX 编码进行压缩.

油水井是人类与油藏地质发生作用的连接点, 因此在油田分析处理业务中, 数据的观测位置是以井号为中心的. 但是由于地质位置具有三维特征, 油藏的形成具有一定的区域性特征, 因此在保存油井静态位置信息时不仅会保留井号视角, 还会存储油井在不同地质层的岩层、物性等信息. 这样, 为了实现对油田事件的多视角分析, 要对只具有井号信息的传感器事件进行地理位置补齐. 当前, 对于我国开发成熟的油田公司, 单个油田的区块划分不会多于 100 个, 了解全面的地质层划分也不会多于 1000 个, 因此可分别构建区块关键字升序列表和地质层关键字升序列表, 通过与井号的关联来实现对地理位置的补充, 为规则中基于位置划分推理范围的条件提供支持.

在到达事件处理模块对原子事件进行关键字和地理位置编码后, 将形成原子事件列表.

### 3.2 推理条件过滤

#### 3.2.1 $\beta$ 网络

为了提高规则匹配的执行效率, 需对规则进行预处理, 将原始规则集转换成高效的  $\beta$  网络. 油田规则集的制定由油田用户来制定, 规则的条件既有合取操作, 也有析取操作, 按照对油

田油藏及生产分析的经验来制定. 可形成形如  $x_1 \wedge x_2 \wedge (y_1 \vee (z_1 \wedge z_2)) \rightarrow A_1$  的规则表达式, 其中  $x_i, y_i, z_i$  代表原子条件,  $A_i$  代表执行的动作. 逻辑运算在经过转换后可形成析取范式, 上述规则的转换过程如下:

$$\begin{aligned} x_1 \wedge x_2 \wedge (y_1 \vee (z_1 \wedge z_2)) \rightarrow A_1 &=> \\ (x_1 \wedge x_2 \wedge y_1) \wedge (x_1 \wedge x_2 \wedge z_1 \wedge z_2) \rightarrow A_1 &=> \\ x_1 \wedge x_2 \wedge y_1 \rightarrow A_1 & \\ x_1 \wedge x_2 \wedge z_1 \wedge z_2 \rightarrow A_1 & \end{aligned}$$

这样, 规则在经过标准化转化后都可形成析取范式, 得到的析取范式可分别形成具有原子规则特征的规则来执行, 具有较高的独立性, 可以用树形结构来表达析取范式得到的两个原子规则, 如图 4.

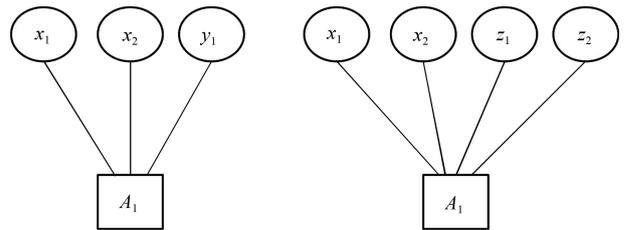


图 4 原子规则树

Fig. 4 Atomic rule trees

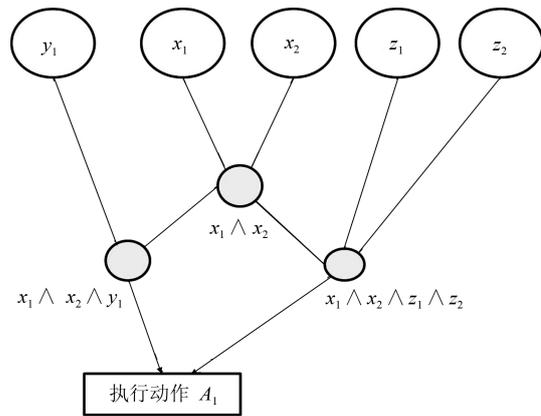


图 5  $\beta$  网络

Fig. 5  $\beta$  network

从上图中的原子规则树可以看出, 如果将规则树中的原子条件按照关键字升序排列, 那么原子条件  $x_1$  和  $x_2$  将重复出现两次, 也意味着事件与条件的匹配将重复进行两次, 降低了匹配效率并占用了更多的规则存储空间. 因此, 对具有重复原子条件的规则进行原子条件合并, 构建  $\beta$  网络, 如图 5 所示.

在  $\beta$  网络中, 作为输入的原子条件只出现一次, 对于多条规则中出现的相同原子条件通过复合条件  $x_1 \wedge x_2, x_1 \wedge x_2 \wedge y_1, x_1 \wedge x_2 \wedge z_1 \wedge z_2$  的形式来实现.

### 3.2.2 基于规则的原子条件筛选

原子条件筛选流程图如图 6 所示.

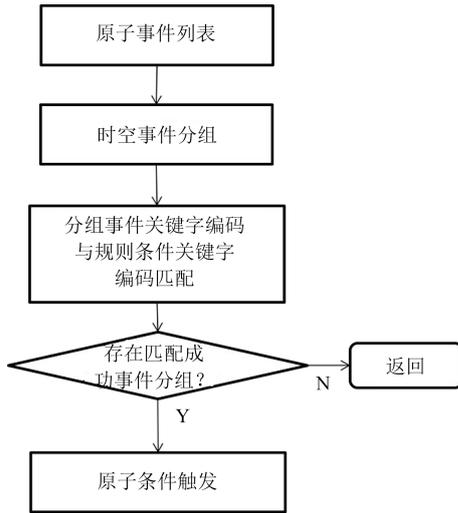


图 6 原子条件筛选流程图

Fig. 6 The flow chart for atomic condition filtering

在油田业务推理过程中, 数据视角以井号为中心, 辅以所属区块、地质分层等位置信息, 所以在形成原子事件列表后, 需要按照空间相关和时间相关性进行事件分类. 如上图所示, 时空事件分组指对原子事件列表中的任意事件的地理位置编码和时间窗口按照用户指定模式进行逻辑与操作, 当且仅当两个事件的地理位置编码和时间窗口编码与用户指定时空条件的逻辑与操作均不为 0 时, 该两个事件可以形成一个分组; 否则将形成新的时空事件分组. 算法说明如下:

#### 算法 1. 时空事件分组

输入. 到达原子事件集  $E$

输出. 带有分组标记的原子事件集  $E$

- 1) 获取到达原子事件集  $E = \{e_1, \dots, e_n\}$
- 2)  $E.groupid = -1$  //将  $E$  中所有事件的分组 id 置为 -1
- 3)  $Groud\_id = 0$  //初始化分组 id
- 4) **if**  $i = 1$  to  $n - 1$  //事件时空分组
- 5) **if**  $e[i].groupid = -1$  **then**
- 6)  $Group\_id = Group\_id + 1$
- 7)  $e[i].groupid = Group\_id$
- 8) **for**  $j = i + 1$  to  $n$
- 9) **if**  $e[i].geo \wedge e[j].geo \neq 0$  and  $e[i].timewidow \wedge e[j].timewidow \neq 0$  **then**
- 10)  $e[j].groupid = e[i].groupid$
- 11) **end if**
- 12) **end for**
- 13) **end if**
- 14) **end for**

然后对同一分组中事件的关键字编码进行逻辑或操作并得到结果  $KG$ , 将  $KG$  与所有规则逐条进行匹配, 只有当匹配结果表明该规则的所有条件关键字均在事件分组中出现, 将事件分组传递到原子条件触发模块, 进行规则执行, 算法说明如下.

#### 算法 2. 分组事件与规则匹配

输入. 同组事件集  $E = \{e_1, \dots, e_n\}$ , 原子规则集  $R = \{r_1, \dots, r_m\}$

输出.  $E$  的继续执行标志  $E.status$  //标志位为 1, 则可继续执行推理

- 1)  $E.status = 0$  //标志位初始化
- 2)  $K_G = B(e[1].key)$
- 3) **for**  $i = 2, \dots, n$  //计算事件集关键字编码逻辑或
- 4)  $K_G = K_G \vee B(e[i].key)$
- 5) **if**  $K_G = 0$  **then** break
- 6) **end if**
- 7) **end for**
- 8) **for**  $j = 1, \dots, m$  //计算分组事件集与规则的相关性
- 9) **if**  $B(r[j].key \wedge B(r[j].key))$  **then**
- 10)  $E.status = 1$
- 11) break
- 12) **end if**
- 13) **end for**
- 14) **return**  $E.status$

算法 2 可以用来判断当前分组事件是否覆盖原子规则中所有条件的关键字, 只有当原子规则条件中各关键字相关事件全部到达, 该规则方可执行. 通过该判断, 可以有效过滤掉不相关事件, 即不相关事件不会进行原子条件触发.

例如: 假设规则集  $R = \{R_1, R_2, R_3\}$ , 其中  $R_{condition.i}$  为  $R_i$  中的规则条件;

$$R_{condition.1} = \{(k_{11}, o_{11}, v_{11}), (k_{21}, o_{21}, v_{21})\}$$

$$R_{condition.2} = \{(k_{12}, o_{12}, v_{12}), (k_{22}, o_{22}, v_{22}), (k_{32}, o_{32}, v_{32})\}$$

$$R_{condition.3} = \{(k_{13}, o_{13}, v_{13}), (k_{23}, o_{23}, v_{23})\}$$

其中,  $k_{ij}$ ,  $o_{ij}$ ,  $v_{ij}$  分别代表第  $j$  个规则条件中的第  $i$  个关键字、操作符和阈值, 且  $k_{im} = k_{in}$  ( $m \neq n$ ), 这样, 可形成规则集  $R$  的基于原子条件关键字的升序列表.

$$L(R_{condition.all}) = \{(k_{11}, o_{11}, v_{11}), (k_{12}, o_{12}, v_{12}), (k_{13}, o_{13}, v_{13}), (k_{21}, o_{21}, v_{21}), (k_{22}, o_{22}, v_{22}), (k_{23}, o_{23}, v_{23}), (k_{32}, o_{32}, v_{32})\}$$

因此, 基于  $L(R_{condition.all})$  的规则条件

Bitmap 编码分别为  $B(\mathbf{R}_{\text{condition}_1}) = (1001000)$ ,  $B(\mathbf{R}_{\text{condition}_2}) = (0100101)$ ,  $B(\mathbf{R}_{\text{condition}_3}) = (0010010)$ .

假设对于到达原子事件时空分组形成的一组事件中包括两个事件  $e_1$  和  $e_2$ , 其中  $e_1 = (k_{1x}, v_1.\text{geo}_1, \text{time\_window}_1)$ ,  $e_2 = (k_{2x}, v_2.\text{geo}_2, \text{time\_window}_2)$ , 事件  $e_1$  的 Bitmap 编码为  $B(e_1) = (1110000)$ ,  $e_2$  的 Bitmap 编码为  $B(e_2) = (0001110)$ , 计算  $KG = B(e_1) \vee B(e_2) = (1111110)$ , 然后分别计算  $KG$  各规则条件编码的逻辑与值, 有

$$KG \wedge B(\mathbf{R}_{\text{condition}_1}) = (1111110) \wedge (1001000) = (1001000) = B(\mathbf{R}_{\text{condition}_1})$$

$$KG \wedge B(\mathbf{R}_{\text{condition}_2}) = (1111110) \wedge (0100101) = (0100100) \neq B(\mathbf{R}_{\text{condition}_1})$$

$$KG \wedge B(\mathbf{R}_{\text{condition}_3}) = (1111110) \wedge (0010010) = (0010010) = B(\mathbf{R}_{\text{condition}_1})$$

油田业务分析过程中, 有些推理执行结果是进行进一步模型计算, 并将结果反馈到推理系统继续使用, 这时, 需要将具有计算特性的推理执行动作结果反馈到到达事件处理模块, 重新进行编码计算, 以触发相应规则的执行.

#### 4 实验与分析

为对实时推理引擎在油田业务中的执行性能进行分析, 本文推理引擎对实际油田业务进行了部署测试.

#### 4.1 数据集与实验平台

本文构建数据集主要包括规则集合和事件集合. 规则集包括对单采油井生产优化推理规则子集、单注水井注水优化推理规则子集、采注联通关系分析规则子集以及在约束条件下采注联合优化措施选取规则子集等. 图 7 所示为油水井联合优化措施流程图.

其中采油单井生产优化推理规则子集中包括 PVT (Pressure-volume-temperature) 模型计算分析规则、多相管流模型计算分析规则、采油井流入动态模型计算分析规则、单采油井最大产量优化设计规则与定产量优化设计规则等; 单注水井优化规则子集中包括吸水能力分析规则、水驱特征分析规则、注水井生产优化设计规则等; 油水井协同优化规则子集包括开发生产系统现状分析规则、递减分析规则、采油井与注水井单井优化备选方案管理规则、物质平衡分析规则、采注井动态连通关系计算分析规则、油水井协同生产组合优化方案选取规则等. 各规则集执行侧将进行模型计算与识别分析功能, 并反馈到推理引擎中进行进一步优化规则的触发. 上述单井优化规则集将形成多种采注优化采注方案组合, 对多组优化采注组合方案, 在系统整体目标约束规则下进行最优组合选取, 系统整体目标通常包括总产量高、总欠注水量低、总生产能耗低等, 阈值设定依据油田生产特征来人工设定. 本文采用对规则集进行了原子规则拆分, 形成测试规则集合.

事件集合采用勘探开发静态数据、万口油水井规模的生产实时工况与动态数据以及 300 口生产参数可调整油水井. 在试验过程中, 通过设定分析需要覆盖的区块, 来控制到达事件数量的变化.

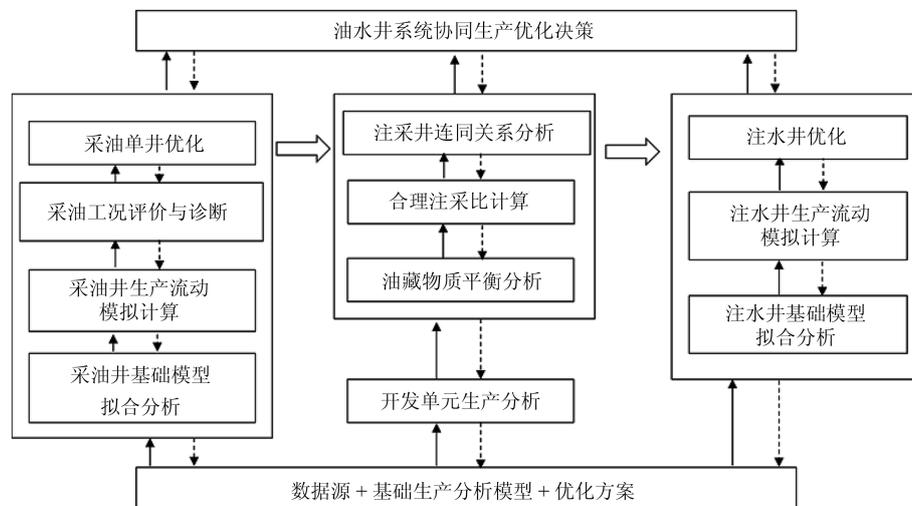


图 7 油水井采注协同优化流程图

Fig. 7 The co-optimization flow chart for injection-production in oil and water wells

实时推理引擎部署在国内某油田公司, 平台硬件配置为 Intel Core i5, 2 GB 内存和百兆以太网卡.

#### 4.2 实验结果及其分析

本文提出的实时推理引擎通过基于 Bitmap 的事件处理来实现对到达事件的筛选, 以优化规则引擎的匹配过程, 目的在于提升对更多到达事件的可处理能力. RETE 作为经典的产生式推理算法已经得到了广泛的研究, LFA 算法作为经典的规则引擎匹配算法也致力于减少模式匹配次数, 因此, 本文选取 RETE 和 LFA 作为性能对比算法. 如图 8 所示为在规则数为 5000 时, 本文方法在到达事件处理能力上与 RETE、LFA 的对比结果, 其中到达事件量为每秒钟的事件数量.

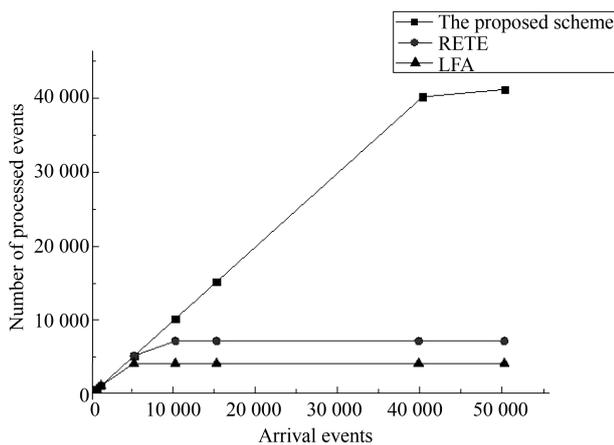


图 8 不同到达事件情况下性能对比图

Fig. 8 The performance comparison with different numbers of arrival events

从图 8 试验对比结果可以看出, 本文实时推理引擎、RETE 推理引擎和 LFA 算法在到达事件量为 4500 events/s 时, 均可对到达事件进行实时处理: 随着同时到达事件量增加至 5000 events/s, 本文方法和 RETE 算法仍可实时处理到达事件, 但是 LFA 算法已经不能实时处理更多的到达事件, 其原因是 LFA 算法在对到达事件的对比方法开销要大于 Bitmap 方法, 加之等待执行的到达事件对资源的耗费, 使得其处理能力难以处理更多的到达事件: 在到达事件数在 5000 events/s 以上时, RETE 算法可实时响应的事件数量没有显著增加, 本文方法在到达事件数量小于 40000/s 的时候, 均可对到达事件进行实时处理.

图 9 所示为在每秒到达事件数为 20000 时, 不同方法在不同原子条件数量下, 推理引擎的可处理事件数量. 可以看出 RETE 和 LFA 算法在原子条件数增加的情况下, 可处理事件数量下降明显, 在原子条件数量达到 20000 时, RETE 算法的可实时处

理事件数维持在 2000 左右, LFA 算法的可实时处理事件数下降到 500 左右; 本文方法在原子条件数达到 10000 时, 可处理事件数对比达到事件数量处理能力略有下降, 其后维持在稳定状态.

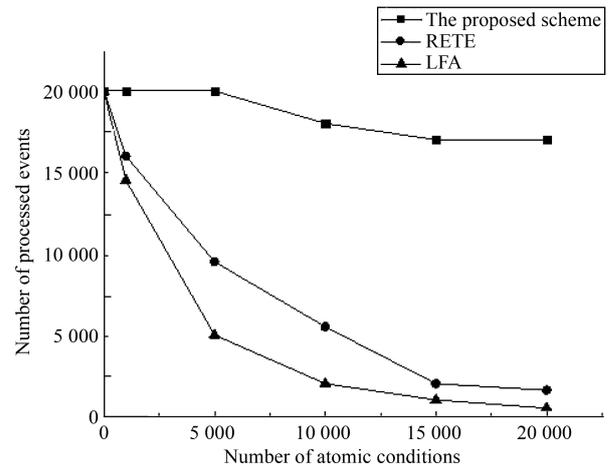


图 9 不同原子条件下性能对比图

Fig. 9 The performance comparison with different numbers of atomic conditions

通过对本文实时推理引擎、传统 RETE 算法以及 LFA 算法的比较可以看出, 基于 Bitmap 的实时推理引擎在筛选无效到达事件的处理以及模式匹配方面的改进是有效的, 使得整体性能优于传统方法.

基于本文油水井采注优化实时推理引擎, 对油田 300 口可调整作业参数的油水井口进行试验, 单井平均日耗电量从 200 kWh 降低到 145 kWh、区块生产时效提高 100 天, 对比应用前具有有效性.

## 5 结论

本文针对油田采注联合优化的实现过程中, 需要大量融入人类经验进行全局分析的场景, 提出了一种基于 Bitmap 编码的实时推理引擎, 通过对产生的事件关键字进行比特位级编码及压缩, 可以有效提升事件与规则的匹配能力. 并针对油田业务规则中的时空相关性, 对产生事件进行筛选, 对当前规则无效事件, 将不会进入规则触发流程, 有效减少了大量到达事件对规则推理引擎的压力, 提升了实时推理引擎的性能. 针对某油田公司业务系统进行了推理引擎实验, 结果表明本文方法在到达事件数和规则数增加时, 均有良好的处理性能. 后续工作将对油井采注优化过程中各参数的选择进行基于大数据分析的智能学习, 以使得推理引擎能更快的适应于不同地质条件的油田场景.

## References

- 1 O'Neil P E. Model 204 architecture and performance. In: Proceedings of the 2nd International Workshop on High

- Performance Transaction Systems. London, UK: Springer, 1987. 40–59
- 2 O'Neil P E, Quass D. Improved query performance with variant indexes. *ACM SIGMOD Record*, 1997, **26**(2): 38–49
  - 3 van Schaik S J, de Moor O. A memory efficient reachability data structure through bit vector compression. In: Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data. Athens, Greece: ACM, 2011. 913–924
  - 4 Antoshenkov G. Byte-aligned bitmap compression. In: Proceedings of the 5th Data Compression Conference. Snowbird, UT, USA: IEEE, 1995. 476
  - 5 Wu K S, Otoo E J, Shoshani A. Optimizing bitmap indices with efficient compression. *ACM Transactions on Database Systems (TODS)*, 2006, **31**(1): 1–38
  - 6 Deliège F, Pedersen T B. Position list word aligned hybrid: optimizing space and performance for compressed bitmaps. In: Proceedings of the 13th International Conference on Extending Database Technology. Lausanne, Switzerland: ACM, 2010. 228–239
  - 7 Fusco F, Stoecklin M P, Vlachos M. NET-FLi: on-the-fly compression, archiving and indexing of streaming network traffic. *Proceedings of the VLDB Endowment*, 2010, **3**(1–2): 1382–1393
  - 8 Kim S, Lee J, Satti S R, Moon B. SBH: super byte-aligned hybrid bitmap compression. *Information Systems*, 2016, **62**: 155–168
  - 9 Wen Y H, Chen Z, Ma G, Cao J W, Zheng W X, Peng G D, Li S W, Huang W L. SECOMPAX: a bitmap index compression algorithm. In: Proceedings of the 23rd International Conference on Computer Communication and Networks (ICCCN). Shanghai, China: IEEE, 2014. 1–7
  - 10 Wen Y H, Wang H, Chen Z, Cao J W, Peng G D, Huang W L, Hu Z W, Zhou J, Guo J H. MASC: a bitmap index encoding algorithm for fast data retrieval. In: Proceedings of the 2016 IEEE International Conference on Communications (ICC). Kuala Lumpur, Malaysia: IEEE, 2016. 1–6
  - 11 Wang Fei-Yue. Software-defined systems and knowledge automation: a parallel paradigm shift from Newton to Merton. *Acta Automatica Sinica*, 2015, **41**(1): 1–8  
(王飞跃. 软件定义的系统与知识自动化: 从牛顿到默顿的平行升华. 自动化学报, 2015, **41**(1): 1–8)
  - 12 Wu Dan-Feng, Zeng Guang-Ping, Yan Jing-Ying. Research on rete algorithm supporting evolution rules engine. *Application Research of Computers*, 2013, **30**(6): 1747–1750  
(武丹凤, 曾广平, 闫京颖. 支持演化规则引擎的 rete 算法研究. 计算机应用研究, 2013, **30**(6): 1747–1750)
  - 13 Forgy C L. Rete: a fast algorithm for the many pattern/many object pattern match problem. *Artificial Intelligence*, 1982, **19**(1): 17–37
  - 14 Hong Y G, Kim H J, Park H D, Kim D H. Adaptive GTS allocation scheme to support QoS and multiple devices in 802.15.4. In: Proceedings of the 11th International Conference on Advanced Communication Technology. Phoenix Park, Ireland: IEEE, 2009. 1697–1702
  - 15 Xu Jiu-Qiang, Lu Suo, Liu Da-Peng, Kong Qiu-Shi. Research on RFID composite events detection based on improved rete algorithm. *Journal of Northeastern University (Natural Science)*, 2012, **33**(6): 806–809, 814  
(徐久强, 卢锁, 刘大鹏, 孔求实. 基于改进 Rete 算法的 RFID 复合事件检测方法. 东北大学学报(自然科学版), 2012, **33**(6): 806–809, 814)
  - 16 Kawakami T, Yoshihisa T, Yanagisawa Y, Tsukamoto M. A rule processing scheme using the rete algorithm in grid topology networks. In: Proceedings of the 29th International Conference on Advanced Information Networking and Applications (AINA). Gwangju, Korea: IEEE, 2015. 674–679
  - 17 Kawakami T, Yoshihisa T, Tsukamoto M. A control method of ubiquitous computers using the RETE algorithm in grid topology network. In: Proceedings of the 3rd Global Conference on Consumer Electronics (GCCE). Tokyo, Japan: IEEE, 2014. 551–552
  - 18 Kawakami T, Yoshihisa T, Fujita N, Tsukamoto M. A rule-based home energy management system using the RETE algorithm. In: Proceedings of the 2nd Global Conference on Consumer Electronics (GCCE). Tokyo, Japan: IEEE, 2013. 162–163
  - 19 Kawakami T, Fujita N, Yoshihisa T, Tsukamoto M. An evaluation and implementation of rule-based home energy management system using the RETE algorithm. *The Scientific World Journal*, 2014, **2014**: Article No. 591478
  - 20 Pallavi M S, Vaisakh P, Reshna N P. Implementation of rete algorithm using course finder system. In: Proceedings of the 2016 International Conference on Data Mining and Advanced Computing (SAPIENCE). Ernakulam, India: IEEE, 2016. 96–100
  - 21 Wu X D. LFA: a linear forward-chaining algorithm for AI production systems. *Expert Systems*, 1993, **10**(4): 237–242
  - 22 Feng Jian-Zhou, Song Sha-Sha, Kong Ling-Fu. Research on semantic association and decision method of the internet of things. *Acta Automatica Sinica*, 2016, **42**(11): 1691–1701  
(冯建周, 宋沙沙, 孔令富. 物联网语义关联和决策方法的研究. 自动化学报, 2016, **42**(11): 1691–1701)
  - 23 Zhu Xiu-Li, Li Long, Li Pan-Chi. Forecasting methods of oil field development indexes based on T-S reasoning networks. *Application Research of Computers*, 2011, **28**(8): 2991–2993  
(朱秀莉, 李龙, 李盼池. 基于 T-S 推理网络的油田开发指标预测方法. 计算机应用研究, 2011, **28**(8): 2991–2993)



刘阳 中国科学院沈阳自动化研究所副研究员. 2011 年获得东北大学计算机应用技术专业博士学位. 主要研究方向为工业物联网数据处理、语义数据处理及智能制造.

E-mail: liuy@sia.cn

(LIU Yang Associate professor at Shenyang Institute of Automation Chinese Academy of Sciences. She received her Ph.D. degree in computer application technology from Northeastern University in 2011. Her research interest covers industrial internet of things data processing, semantic data processing and intelligent manufacturing.)

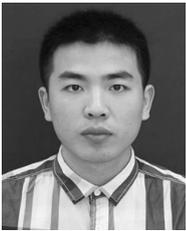


**张天石** 中国科学院沈阳自动化研究所助理研究员. 2013 年获得北京邮电大学自动化学院硕士学位. 主要研究方向为智能优化算法以及工业物联网本体设计.

E-mail: zhangtianshi@sia.cn

**(ZHANG Tian-Shi** Assistant professor at Shenyang Institute of Automation Chinese Academy of Sciences. He

received his master degree from School of Automation, Beijing University of Post and Communication in 2013. His research interest covers intelligence optimization algorithm and ontology design in industrial things.)



**李世超** 中国科学院沈阳自动化研究所助理研究员. 2014 年获得东北大学信息学院硕士学位. 主要研究方向为油田优化开采以及智慧油田应用.

E-mail: lishichao@sia.cn

**(LI Shi-Chao** Assistant professor at Shenyang Institute of Automation, Chinese Academy of Sciences. He received

his master degree from the School of Information, Northeastern University in 2014. His research interest covers optimized exploitation of oilfield and application of intelligent oilfield.)



**佟星** 中国科学院沈阳自动化研究所助理研究员. 2012 年获得哈尔滨工业大学计算机学院硕士学位. 主要研究方向为自然语言处理以及工业物联网本体设计. E-mail: tongxing@sia.cn

**(TONG Xing** Assistant professor at Shenyang Institute of Automation, Chinese Academy of Sciences. He received

his master degree from the School of Computer, Harbin In-

stitute of Technology in 2012. His research interest covers natural language processing and ontology design in industrial things.)



**曾鹏** 中国科学院沈阳自动化研究所研究员. 2005 年中国科学院大学机械电子工程专业博士学位. 主要研究方向为工业无线传感器网络, 智能电网以及需求响应. 本文通信作者.

E-mail: zp@sia.cn

**(ZENG Peng** Professor at Shenyang Institute of Automation, Chinese Academy of Sciences. He received his Ph.D. degree

in mechatronic engineering from Graduate School of the Chinese Academy of Sciences in 2005. His research interest covers wireless sensor networks for industrial automation, smart grids, and demand response. Corresponding author of this paper.)



**于海斌** 中国科学院沈阳自动化研究所研究员. 1997 年获得东北大学控制理论与控制工程专业博士学位. 主要研究方向为自动化控制系统, 先进制造技术和智能电网的基础与应用研究.

E-mail: yhb@sia.cn

**(YU Hai-Bin** Professor at Shenyang Institute of Automation, Chinese Academy of Sciences. He received his Ph.D. degree

in control theory and control engineering from Northeastern University in 1997. His research interest covers basic and applied research in the areas of automation control systems, advanced manufacturing techniques and smart grids.)