使用上下文无关位置文法解析超图

Gennaro Costagliola*萨莱诺大学 gencos@unisa.it

Federico Vastarini†萨莱诺大学 fvastarini@unisa.it

我们介绍了一种针对由上下文无关超边替换文法生成的超图进行解析的新型、仍在开发中的方法。该方法基于一种面向位置文法的新型 LR 语法分析技术，目前也在积极开发中。我们的方法核心是将超边替换转换为带有额外结构约束的位置文法，从而能够使用基于排列的操作，确定产生式右部超边的正确顺序。初步结果还揭示了图生成中的歧义与图识别中的歧义之间的区别。尽管该方法能够解析的超边替换语言的确切类别仍在研究中，但这种方法为今后推广到表达能力更强的文法形式体系奠定了良好基础。图解析在众多领域仍是一个普遍相关的问题，我们的工作旨在推进对这一挑战的理论和实践理解。

引言：在上下文无关的 [2, 6] 超边替换语言（HRL）[7, 11] 中解析超图，是一个已得到充分研究但计算上颇具挑战的问题。尽管 [8], 对这一问题进行了深入探讨，并将预测移进-归约（PSR）分析器作为 SLR(1) 字符串语法分析的图类比方法加以引入，但这项尚在进行中的工作提出了一种新方法，旨在为一类目前仍在研究中的此类语言构建高效解析器。其核心思路是利用产生式右部超边的顺序；这一思路最初由 [14] 为一致图生成而提出，现将其调整后应用于解析领域。该方法分为两部分。首先，将超边替换文法（HRG）转换为位置文法（PG） [5, 3]。随后，提出一种方法，对所得文法进行重构和重新排序，以期消除语法分析冲突并实现高效解析。该方法最终得到一个基于项目的解析器 [10] ，并使用点运算符。表示超图的位置字符串采用经过调整的 LR 技术进行解析 [12]。我们提出的方法基于自底向上分析器，尚未考虑自顶向下分析器；相关讨论见 [9], 。另一种相关方法则针对另一类超图使用节点排序，见 [1]。

超边替换文法：我们首先简要介绍上下文无关超边替换文法形式体系所需的基础知识，以确保在整个转换过程中保留关键属性。关于超边替换语言的全面介绍见 [13, 7, 11]。带标记超图是标准图的一种推广，其中每条边（称为超边）都带有一个标签。形式上，设 $\mathrm{type} : C \to \mathbb{N}_0$ 是固定标记集 C 上的类型函数，则 C 上的超图是一个元组 $H = (V_H, E_H, \mathrm{att}_H, \mathrm{lab}_H, \mathrm{ext}_H)$，其中 V_H 是有限顶点集，E_H 是有限超边集，

*G. Costagliola 开放研究者与贡献者身份标识符：0000-0003-3816-7765

†F. Vastarini 开放研究者与贡献者身份标识符：0000-0002-8002-5704（通讯作者）作者按字母顺序排列。

L. Lambers 和 O. Semer´ath（编）：GCM 2025 EPTCS440, 2026 年，第 1–5 页，doi:10.4204/EPTCS.440.1

© Gennaro Costagliola、Federico Vastarini 本作品依据知识共享署名许可协议授权。

$att_H: E_H \to V_H^*$ 是一个映射，为每个 $e \in E_H$ 指定一个附着节点序列；$lab_H: E_H \to C$ 是一个将每条超边映射到一个标签的函数，使得 $type(lab_H(e)) = |att_H(e)|$；$ext_H \in V_H^*$ 是一个两两不同的外部节点序列。请注意，外部节点仅在超图的推导过程中使用。我们用 type(H) 表示 $|ext_H|$。附着序列 $|att_H(e)|$ 的长度即为 e 的类型。在本文中，我们主要关注不含孤立节点或超边的超图。也就是说，每条超边的类型至少为 1，且每个节点都必须至少有一条邻接边。外部节点用实心圆表示。将超边替换为类型相匹配的超图所需的基本操作由产生式定义。设 $N \subseteq C$ 是非终结标签的一个子集。那么，$p = (A, R)$ 是 N 上的一个产生式，其中 $lhs(p) = A \in N$ 是被替换超边的标签，$rhs(p) = R$ 是一个满足 $type(R) = type(A)$ 的超图。我们用小写字母表示终结超边，用大写字母表示非终结超边。

给定一个有序集合 $\{\alpha_1, \ldots, \alpha_n\}$，其中当 $i < j$ 时 $\alpha_i < \alpha_j \in \mathbb{N}$，我们将超边替换文法（即 HRG）定义为一个元组 $G = (N, \Sigma, P, S, (mark_p)_{p \in P})$，其中 $N \subseteq C$ 是有限的非终结标签集合，$\Sigma \subseteq C$ 是有限的终结标签集合，且 $N \cap \Sigma = \emptyset$；P 是有限的产生式集合，$S \in N$ 是起始符号，$(mark_p)_{p \in P}$ 是一个函数族 $mark_p: E_R \to \{\alpha_1, \ldots, \alpha_n\}$，用于为产生式 p 的右部中的每条超边分配一个标记。对于每一对 e_i、$e_j \in E_R$，当 i , j 时，$mark(e_i)$, $mark(e_j)$。这种顺序被视为文法的一项属性，因此具有不同顺序的文法被视为不同的文法。始终可以为 HRG 产生式右部中的超边任意指定顺序，而我们的方法旨在寻找一个顺序（如果存在），使得能够生成良构文法。

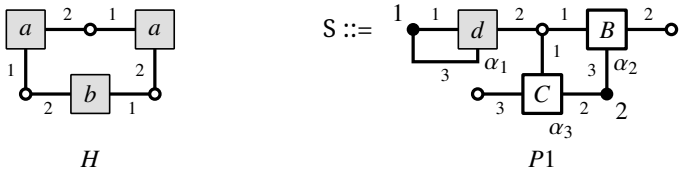


图1：超图和一条超边替换文法产生式的示例

图1中的超图 H 是一个循环图示例，其中包含三条标签分别为 a、a 和 b 的终结型 2 超边。去除标记和外部节点后，该超图便是解析器输入的一个示例。请注意，例如，循环图没有可自动识别的起始点。同一图中的产生式 P1 描述了将左部带有标签 S 的超边，替换为右部中同类型超图的过程。右部的 2 型超图有 5 个节点，其中 2 个以实心圆表示的节点是外部节点。带有标签 d 的超边是终结超边，其余带有标签 B 和 C 的超边则是非终结符。标记 α_1、α_2 和 α_3 表示解析过程中考虑超边的一种可能顺序。共有 n! 种可能的顺序，其中 n 是超边的数量。

从超边替换文法到位置文法：我们简要介绍上下文无关位置文法，并描述如何在这一框架中表示超边替换文法。位置字符串 $s = c_1 e_1 \ldots c_n e_n$ 是由元素组成的有序序列，每个元素前面都有一个连接符合取式。我们将连接符合取式之后的元素称为其目标。每个元素由一个符号和一系列接口构成。合取式中的每个连接符描述源元素的一个接口与目标元素的一个接口之间的二元关系。这类关系也可能涉及同一元素的两个不同接口。连接符表示为 $k^z\, l$，其中 k 是源的接口，l 是目标的接口。为了

确保与现有记法兼容，我们将 z 定义为源相对于目标的距离减一。也就是说，$z = -1$ 表示源与目标重合，而 $z = 0$ 表示源指向紧邻目标之前的元素，依此类推。按照约定，默认采用 $z = 0$，也可以省略。为了表示连接到单个超边的节点，我们还使用一种一元关系，该关系仅涉及单个元素的一个接口。例如，图1中 P1 的右部里带有 B 标签的超边的第二个连接点。为了符合解析器当前输入的形式，合取式中不会显式显示一元关系，而是根据文法产生式右部所使用元素的描述推断出来。例如，图1中的产生式对应于以下位置产生式：

$S \Rightarrow 1^{-1}_3\ d\ 2\ \underline{1}\ C\ \ 2^{1}_1 \wedge 3_2\ B\ (S\ 1 = d1, S\ 2 = C2)$

为了使用 PG 表示超边替换文法，我们考虑满足以下性质的可逆关系：

▹ 唯一性：每个接口必须且只能参与一种类型的关系。▹ 对称性：二元关系是对称的：如果接口 1 与接口 2 之间存在关系，那么接口 2 与接口 1 之间也存在相同的关系。▹ 传递性：二元关系具有传递性：如果接口 1 与接口 2 之间存在关系，且接口 2 与另一个接口 3 之间存在相同的关系，那么接口 1 与接口 3 之间也必须存在这种关系。

为了描述超图的结构，我们使用二元关系“共享”来表示两种情况：两个不同的超边与同一节点相邻，或者同一超边的两个不同附着点连接到同一节点。当两个或更多超边与同一节点相邻时，就形成一种同时具有对称性和传递性的关系。同一超边的两个或更多附着点连接到同一节点时，情况也一样。唯一性限制确保一个接口可以参与二元关系或一元关系，但不能同时参与两者。例如，图1中的超图对应以下位置字符串：$a\ 2\ 1\ a\ \ 1^{1}_2 \wedge 2_1\ b$。

解析器：该解析器遵循 pLR解析方法所依据的原则，如 [5, 3] 中所述。这是对 LR解析 的扩展，其 LR项 的构造采用位置字符串，而非传统的符号字符串。项目中的带点符号表示预期读取的符号，而带点连接符合取式则表示应从何处获取下一个符号。该解析方法适用于格式良好的位置文法，也就是说，这些文法遵守 [5], 中定义的限制：对于每个产生式，左部非终结符的每个输入连接点都与右部最左符号的一个连接点相连。 [8], 中提出的基于LR的解析方法也有类似要求，文法设计者必须仔细安排产生式顺序，以确保解析器正常运行。与这种方法相比，pLR 解析方法还有其他区别，例如，如 [4], 所示，它能保证项目集的构造过程始终有限，从而实现对结构化流程图的基于LR的解析器的构造。

我们提出的方法通过引入一种文法转换过程，能够正确调整产生式中符号的顺序，从而不再要求输入必须是良构文法。这一过程的适用范围目前仍在研究中，既可以作为方法的初始步骤，也可以在创建项目集时应用，因此语言设计者在定义文法时拥有更大的灵活性。它可视为通向超边替换语言确定性 LR 风格解析的 [8] PSR 方法的一条补充路径。PSR 保留了 HRG 形式体系，并通过可行前缀和自由边选择性质等全局条件来确保确定性；而我们的方法则将超边替换文法转换为位置形式，并通过

对产生式进行适当排列来确保确定性。这两种视角共同凸显了基于字符串的 LR 技术与其基于图的推广之间日益增强的趋同。

为说明我们的解析方法，我们引入一个基于环图文法的示例。以下位置文法对应于图 2 所示的环图超边替换文法，其中元素的顺序反映了超边的顺序。

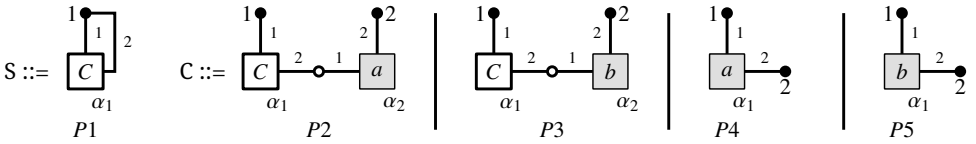


图 2：环图超边替换文法

$S \longrightarrow 2^{-1}_1\ C\ (S1 = C1)\ C \longrightarrow C\ 2\ 1\ \underline{a}\ (C1 = C1, C2 = a2)\ C \longrightarrow C\ 2\ 1\ b\ (C1 = C1, C2 = b2)$

$C \longrightarrow a\ (C1 = a1, C2 = a2)\ C \longrightarrow b\ (C1 = b1, C2 = b2)$

给定一个位置文法（PG），可以通过对其产生式应用一系列置换，得到一个等价的良构文法：

- 简单置换：重新排列某个产生式右部元素序列的顺序。由于改变右部元素的顺序会得到一个表示不同顺序的同构超图，因此只会影响解析过程中字符串的遍历方式。

- 重复置换：与前一种情况一样重新排列元素，但同时在文法中保留原产生式的一个副本。这种方法确保产生式可以按两种顺序处理，但代价是文法规模增大。

上述位置文法是一个良构文法的示例，无论从哪个元素开始解析，它都能为环图生成高效且无冲突的 pLR 解析器。在我们的方法中，对产生式进行适当置换时，超边的顺序发挥着类似于 PSR 解析中 Follow 和 Follow* 集合的作用，引导解析器做出确定性选择。此外，正如 PSR 依靠自由边选择性质来确保任何符合条件的边都能无失败地移入，我们使用置换也能确保无论解析器首先遇到哪个超边，推导都可以在不回溯的情况下继续进行。

例如，若将 P1 置换为以下形式，循环图位置文法（PG）便不再满足良构性：

$C \longrightarrow a\ 1_2\ C\ \ (C1 = C1, C2 = a2)$

在这种情况下，产生式左部非终结符的进入附着点 1，并未与产生式右部最左符号 a 的任何附着点相连。PSR 框架中与之相关的概念是可行前缀。在我们的设定中，良构性发挥着类似作用，确保前缀不会留下悬空或未锚定的附着点。

结论：本研究目前的目标主要聚焦于以下几个方面：

高效算法：由于解析器的行为会直接受到元素顺序的影响，因此，设计一种能够尽可能减少重复置换数量的高效算法，对于实现高效解析至关重要。理想情况下，应通过分析文法结构来确定正确的顺序，识别外部节点和附着点如何在各个产生式中引入并传播。借助此类分析，解析器便可系统地确定哪些超边应优先出现在右部，从而确保良构性，而无需任意或穷举式地重新排序。

HRL 类：另一种可能的扩展是借助 PG，研究这种解析方法适用的 HRL 类。

参考文献

[1] Henrik Bj¨orklund、Frank Drewes、Petter Ericson 和 Florian Starke（2021）：超边替换文法的统一解析。《计算机与系统科学杂志》118，第 1–27 页，doi:10.1016/j.jcss. 2020.10.002。[2] Noam Chomsky（1959）：关于文法的某些形式性质。《信息与控制》2(2)，第 137–167 页，doi:10.1016/S0019-9958(59)90362-6。[3] Gennaro Costagliola、Vincenzo Deufemia、Filomena Ferrucci 和 Carmine Gravino（2020）：视觉语言解析中的运行时冲突检测。《计算机语言杂志》57(100943)，第 1–13 页，doi:10.1016/ j.cola.2020.100943。[4] Gennaro Costagliola、Andrea De Lucia 和 Sergio Orefice（1994）：迈向图示语言的高效解析。载于：《高级视觉界面研讨会论文集》（AVI'94），AVI'94，计算机协会，第 162–171 页，doi:10.1145/192309.192344。[5] Gennaro Costagliola、Andrea De Lucia、Sergio Orefice 和 Genoveffa Tortora（1997）：视觉系统实现的解析方法论。《IEEE软件工程汇刊》23(12)，第 777–799 页，doi:10.1109/32.637392。[6]Bruno Courcelle（1987）：上下文无关重写的公理化定义及其在 NLC 图文法中的应用。《理论计算机科学》55(2)，第 141–181 页，doi:10.1016/ 0304-3975(87)90102-2。[7] Frank Drewes、Annegret Habel 和 Hans-J¨org Kreowski（1997）：超边替换图文法。载于：《图文法与图变换计算手册》，第 1 卷，World Scientific，第 95–162 页，doi:10.1142/3303。[8] Frank Drewes、Berthold Hoffmann 和 Mark Minas（2019）：基于超边替换的图文法预测式移进-归约解析器的形式化与正确性。《逻辑与代数方法编程杂志》104，第 303–341 页，doi:10.1016/j.jlamp.2018.12.006。[9] Frank Drewes、Berthold Hoffmann 和 Mark Minas（2020）：作为图变换的图解析。载于：第 13 届图变换国际会议论文集（ICGT20），LNCS12150，Springer，第 221–238 页，doi:10.1007/978-3-030-51372-6_13。[10] Jay Earley（1970）：一种高效的上下文无关解析算法。《ACM杂志》13(2)，第 94—-102 页，doi:10.1145/362007.362035。[11] Joost Engelfriet（1997）：上下文无关图文法。载于：《形式语言手册》第 3 卷，Springer，第 125–213 页，doi:10.1007/978-3-642-59126-6_3。[12] Donald E. Knuth（1965）：从左向右翻译语言。《信息与控制》8(6)，第 607–639 页，doi:10.1016/S0019-9958(65)90426-2。[13] Federico Vastarini（2024年）：上下文无关图语言中的随机图生成。博士论文，约克大学。可访问：https://etheses.whiterose.ac.uk/id/eprint/34974/。[14] Federico Vastarini 和 Detlef Plump（2024年）：上下文无关图语言中的随机图生成。载于：第 13 届计算模型进展国际研讨会论文集（DCM2023），《理论计算机科学电子会议论文集》408，开放出版协会，第 107–120 页，doi:10. 4204/EPTCS.408.7。