重新审视力导向图布局：一种基于 t 分布的新力

Fahai Zhong、Ming liang Xue、Jian Zhang、张帆、Rui Ban、Oliver Deussen 和 Yunhai Wan g

摘要——本文提出 t-FDP 模型，这是一种力导向布局方法，其基于一种由学生t分布定义的新型有界短程力（t 力）。我们的公式具有灵活性，对邻近节点施加有限的排斥力，并可分别调整其短程和长程效应。在力导向图布局中使用此类力，相较于现有方法能够更好地保持邻域，同时维持较低的应力误差。我们采用快速傅里叶变换实现的高效算法，速度比最先进的方法快一个数量级，在图形处理器上则快两个数量级，因此能够通过实时全局和局部调整 t 力，为复杂图进行参数调优。我们通过与最先进方法及其用于交互式探索的扩展进行数值评估，展示了我们方法的优越性。

索引词—图布局，力导向布局，学生t分布，快速傅里叶变换

arXiv:2303.03964v1 [cs.GR] 5 Mar 2023

图 RAPHS 是一种无处不在的数据形式，只要需要表示对象及其关系，就会用到图；人们为寻找有效的可视化方法付出了大量努力。可以说，图最常见的表示形式之一是节点-连线图，它直观且能有效支持各种图分析任务 [3]。

尽管存在许多其他技术，力导向布局（FDP）方法 [4] 仍因能够自动生成节点-连线图布局而备受关注。在这种方法中，图被建模为一个由物体构成、物体之间相互作用的物理系统；布局通过最小化系统的总能量来计算。虽然已有研究提出了若干种力的变体 [4], , 弹簧-电力模型 [1], [5] 仍是最流行的方法。该模型在相连节点之间设置类似弹簧的吸引力，并在所有节点对之间设置电斥力，以使节点保持一定距离。对于大多数图而言，该方法都能生成效果出色的布局 [6], , 尤其是在采用其多层版本——可扩展力导向布局（SFDP） [7]时。由于直观且易于实现，该方法是许多可视化系统（例如 Gephi [8] 和 D3 [9]）的关键组成部分。

弹簧-电力模型的一个典型问题是，每对彼此靠近且相连的节点之间都会产生很大的电学排斥力，而吸引弹簧力却很小。在物理模拟中，这是合理的，因为真实物体之间的接触排斥力应当极其大。然而，这种力可能会破坏图结构，使图中相连的节点在布局中无法成为邻居（见图1(a)底部的统计数据）。尽管近期提出的图布局方法（如 tsNET [2]）能够更好地保留二环邻域，但在有效维持较低的应力误差和一环邻域方面表现不佳（见图1(b)）。DRGraph [10] 提高了 tsNET 的可扩展性，但同样存在应力误差较大的问题。

本文重新审视了弹簧-电力模型，并提出一种基于 t 分布的新型重尾力（简称 t 力），旨在更好地保留局部图结构，同时维持模型的直观性和简洁性。偏离传统的弹簧-电力模型，使我们能够灵活探索吸引力与排斥力之间的相互作用。具体做法是将力分为短程和长程分量，并改进其短程特性。t 力在短程范围内有上界，在长程范围内的表现则类似于电斥力。短程吸引力保持不变。在 FDP 中使用 t 力定义排斥力，并将 t 力与弹簧力结合来定义吸引力，我们得以构建一种全新的基于 t 力的 FDP 模型（简称 *t-FDP* ），同时更好地保留图邻域、节点距离和簇（见图1(c)）。

作为 t-SNE梯度的两个组成部分之一，t 力类似于 t-SNE模型的排斥力 [11],，而 t-SNE 是一种广泛使用的降维方法。这使我们能够利用基于插值的加速策略，通过快速傅里叶变换 [12]加速 t-SNE；该策略可在图形处理器上实现。由于快速傅里叶变换具有并行特性，我们的整个模型都能在 GPU 上高效实现，从而支持交互式参数优化。为了让更多用户能够使用我们的方法，我们提供了一个可直接替换使用的力实现，可用于“d3-force”库。

我们使用 50 个图评估了该方法，这些图的节点数从 72 到 400 万不等，边数从 75 到 1 亿不等。我们对

• *Fahai Zhong*、*Mingliang Xue* 和王云海任职于中国山东大学计算机科学系。电子邮箱：{*zhongfahai*、*xml95007*、*cloudseawang*}*@gmail.com*

• *Jian Zhang* 任职于中国科学院计算机网络信息中心（中国北京）。电子邮箱：*zhangjian@sccas.cn*

• 张帆任职于中国 *SDTBU* 计算机科学与技术学院。电子邮箱：*zhangfan @sdtbu.edu.cn*

• *Rui Ban* 任职于中国北京 *CITC* 智能网络设计研究院。电子邮箱：*b anrui1@chinaunicom.cn*

• *Oliver Deussen* 任职于德国康斯坦茨大学计算机与信息科学系（康斯坦茨）。电子邮箱：*oliver.deussen@uni-konstanz.de*

• 王云海为通讯作者

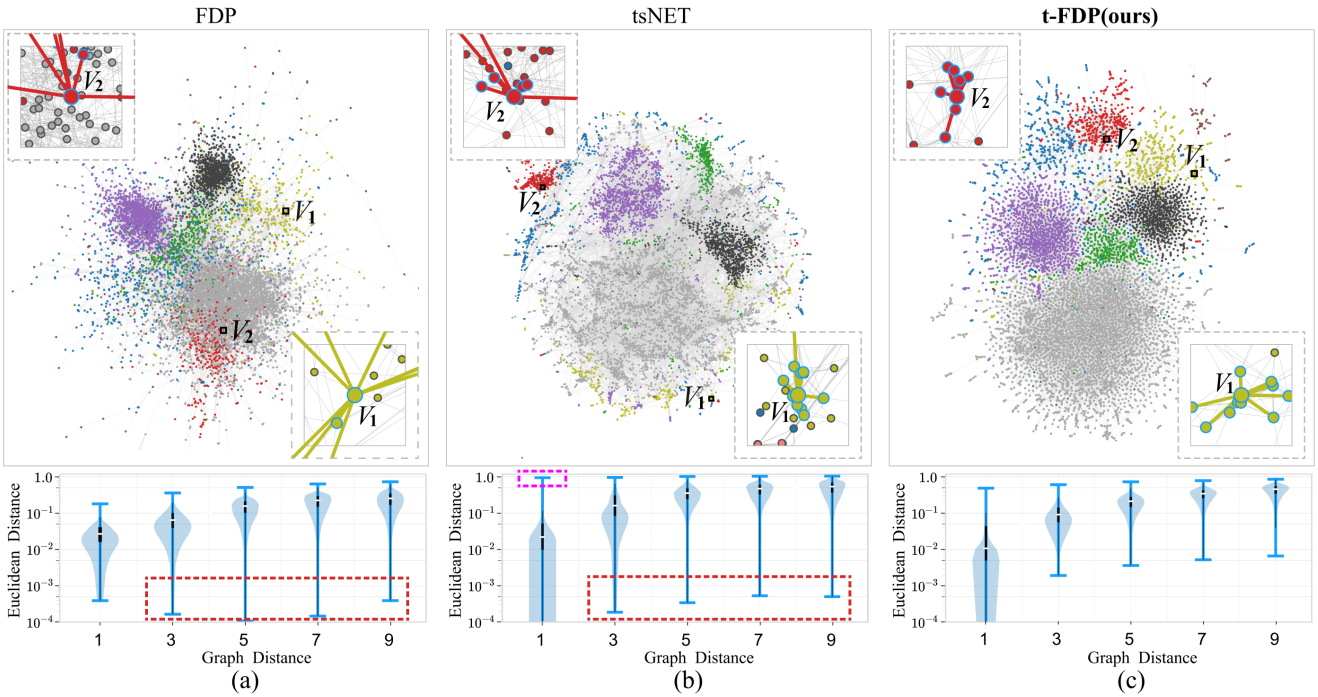


图 1. 通过小提琴图总结图节点在布局空间中的归一化欧氏距离（显示于 (a)、(b)、(c) 底部），将我们的方法（t-FDP）与标准力导向布局（FDP） [1] 以及基于 tsNET [2], 这一先进邻域嵌入方法进行比较，数据来自带标签的合著者网络。FDP 会将图距离较大的节点放置得彼此接近（如 (a) 小提琴图中的红框所示）；tsNET 在一定程度上避免了这一问题（如 (b) 小提琴图中的红框所示），但通常会将图距离较小的节点彼此远置（如 (b) 小提琴图中的粉框所示）。(c) 中的 t-FDP 能够同时实现：对于数据空间中相近的数据点，布局距离较短；对于相距较远的数据点，布局距离较大。因此，它能够清晰地区分全部八个类别，而其他方法则会使部分类别相互混杂。两个示例邻域体现了这一点：在 (a) 中，V_1 的邻居与 V_1 没有连接，并且 V_2 的图邻居大多相距较远。tsNET 在保持 V_1的局部邻域方面表现更好，但 V_2 的邻域具有较大的图距离。在这两种情况下，t-FDP 都能实现更好的邻域保持。

结果质量进行了定量测量，采用了多种结构指标和可读性指标。对于所测试的数据，除应力误差外，我们的方法在所有指标上的表现均与现有方法相当，甚至优于它们；在应力误差方面，我们的方法仅次于应力模型，排名第二。就性能而言，在配备 Intel i7-8700 CPU 的台式计算机上，t-FDP 的 CPU版本比 SFDP 和 DRGraph 快一个数量级，同时显著减少了内存占用。我们的基于GPU的CUDA实现还可以将性能进一步提升一个数量级。

此外，我们提出了 t-FDP 的两种扩展，以便更好地探索图结构。首先，我们展示了如何通过对现有布局重新施加不同强度的排斥力，揭示不同尺度上的结构。其次，我们允许用户通过交互式调整力参数，利用鱼眼式可视化探索感兴趣的节点。由于我们的方法速度很快，即使面对更大的图，也能保持交互性。

总之，我们的主要贡献如下：

- 我们重新审视了弹簧-电力模型，并说明它无法在图中恰当地捕捉局部邻域和聚类结构；
- 我们提出了一种基于 t-分布的新型短程力，并用它构建了新的 FDP 模型，克服了现有 FDP 模型在邻域保留方面的不足；
- 我们提供了基于FFT的快速实现[1] 和该方法的交互式演示[2]，并通过定量评估和扩展展示了其有效性。

1. https://github.com/Ideas-Laboratory/t-fdp
2. https://t-fdp.github.io

2 相关工作

相关工作分为两类：力导向图布局，以及 t 分布及其在图布局中的应用。

2.1 力导向图布局

自 Tutte 首次提出以来， [13], 人们开发了许多力导向布局的变体 [4], [14]。其中，弹簧-电力模型和应力模型是最受欢迎的两类。

弹簧-电力模型。该模型将节点表示为环状接头，将边表示为弹簧， [1],[5] 利用吸引力将相邻节点拉到一起，并利用排斥力使所有节点彼此远离。当系统能量达到最小值时，便得到最优布局。为避免远距离作用力过强，Eades [5]引入了对数弹簧力，以及与距离平方成反比的排斥力。为生成边长均匀的布局，Fruchterman 和 Reingold [1](FR)将吸引力重新定义为与距离平方成正比，并将排斥力定义为与距离成反比。Hu 等人 [7]将排斥力建模为如下形式的幂函数 $f(d) = d^{-p}$，其中 p 为任意正整数；他们发现，对于大多数图，与距离平方成反比的力效果良好。Noack [15] 将这一思路推广到 LinLog 能量模型，该模型采用恒定吸引力以及与距离成反比的排斥力，在所测试的图中形成了分离良好的簇。

作为 LinLog 和 FR模型之间的一种折中，ForceAtlas2 [16]将吸引力设为与距离成正比，并将排斥力设为与距离成反比，以便更好地展示

局部邻域和聚类结构。然而，正如 Fruchterman 和 Reingold [1]，所指出的，这种配置对于复杂图可能效果不佳，因为强烈的短程排斥力往往会使相应的优化陷入局部极小值。为缓解这一问题，Kumar 等 [17] 提出了一种启发式方法，对排斥力设置上限，将其作为可视化有向无环图的停止条件。相比之下，我们的 t-FDP 模型中的短程 t 力能够在很大程度上缓解一般图中的这一问题（见第 3.3 节）。

为了可视化大型图，人们采用了各种多层方法来提升可扩展性。例如，Barnes-Hut 技术 [18] 和快速多极子方法 [19] 常用于近似长程斥力 [7],，其总体时间复杂度为$O(n \log n)$（针对 n 个节点）。由于 Barnes-Hut 近似易于实现，因此许多 FDP 算法都采用了它，例如 SFDP [7]和 ForceAtlas2 [16]。Gove [20] 提出了随机顶点采样来计算排斥力。该方法生成的布局与 Barnes-Hut 和 FMM 类似，但时间复杂度为 $O(n)$。基于快速傅里叶变换加速，我们的 t-FDP 模型的运行时间也为 $O(n)$，但生成的布局应力误差更低。

应力模型。另外，最初由 Kamada 和 Kawai [21] 提出的应力模型会在每对节点之间关联一根弹簧，其理想长度与两者之间最短路径的长度成正比。这样，该模型生成的布局相比弹簧-电力模型具有更好的全局结构。然而，应力模型更重视相距较远的节点对，因此通常难以保留局部结构。

为了更好地保留此类局部结构，应力模型的局部版本 [6], [22]–[24]已被提出。最大熵应力模型 [6] 仅对指定长度内的边施加应力，但对所有节点施加电斥力。这降低了计算所有节点对最短路径的计算成本。与此相对，t-FDP 旨在通过更好地平衡局部结构和全局结构的表示来改进弹簧-电力模型。我们的实验结果表明，在表示全局结构方面，它优于最大熵应力模型，同时速度也显著更快。

2.2 t 分布及其在图布局中的应用

与高斯分布相比，t 分布 [25] 是一种具有较重尾部的对称钟形分布。该函数定义为：

$$f(x) = \left(1 + \frac{x^2}{2v-1}\right)^{-v},$$

其中，v 是自由度。随着 v 的增加，该分布越发接近高斯分布。

通过采用这种分布来建模嵌入空间中两点之间的相似性，邻域嵌入 t-SNE [11] 不仅极大缓解了降维（DR）的拥挤问题，还能有效保留局部结构。

已有多种基于 t 分布的降维方法被提出，例如 LargeVis [26], 、UMAP [27] 和 TriMap [28] ，旨在保留更多全局结构。另一方面，为提高 t-SNE 的可扩展性，人们开发了多种加速策略，例如 Barnes-Hut 近似 [29], 、基于 GPU 的纹理溅射 [30] 和基于插值的快速傅里叶变换 [12]。其中，基于插值的快速傅里叶变换方法通过使用快速傅里叶变换来近似排斥力（负梯度中的一项），从而实现近线性复杂度。

近年来，基于 t 分布的降维方法已被用于图布局。一种具有代表性的方法是 tsNET [2], ，它采用定制的 t-SNE 来捕捉局部结构。与基于多维尺度分析 MDS [31], 的应力模型相比，tsNET 能为更多种类的图生成高质量布局。为处理大型图，Zhu 等人提出了 DRGraph [10] ，通过使用负采样进行梯度估计，进一步提升了 tsNET 的可扩展性。

秉持类似的思路，我们使用 t 分布来定义我们的 t-FDP 模型中的有界短程力。此外，这种形式使我们能够采用快速傅里叶变换 [12] 来提升可扩展性；其速度在中央处理器上比 DRGraph 快约一个数量级，在图形处理器上则快两个数量级。

3 方法

给定一个包含 n 个节点和 m 条边的无向图，图布局的主要目标是为每个节点计算一个低维位置，使其满足一系列结构与美学标准，包括节点均匀分布、边长均匀以及反映对称性。然而，力导向布局算法，尤其是弹簧-电力模型，并不明确致力于实现这些目标，而仅基于 Fruchterman 和 Reingold [1] 提出的两项设计原则：

P1: 由边相连的节点应绘制得彼此靠近；并且

P2: 一般而言，节点之间不应绘制得过于靠近。

然而，对于包含两个以上节点的图，仅凭这两项原则无法确保相连节点在布局中成为最近邻（见图1(a)）。这会导致重要的局部图结构丢失。为解决这一问题，我们提出第三项设计原则：

P3: 由边相连的节点应绘制得比
未相连的节点彼此更近。这样可以增强局部结构，并更忠实地呈现局部连接模式。

3.1 重新审视弹簧-电力模型

为满足 **P1** 和 **P2**，弹簧-电力模型采用吸引弹簧力 f^a 将相连节点拉近，并采用电排斥力 f^r 使节点彼此排斥。迄今为止，人们使用了不同的函数来定义吸引力和排斥力 [7], [16], 其中许多函数可以写成幂函数的形式：

$$F^a(i,j) = \alpha||\mathbf{x}_i - \mathbf{x}_j||^p, \quad i \leftrightarrow j, \tag{1}$$

$$F^r(i,j) = -||\mathbf{x}_i - \mathbf{x}_j||^{-q}, \quad i \neq j, \tag{2}$$

其中，$\mathbf{x}_i$ 是图节点 i 在布局空间中的二维位置，α 是权重，并且 p 和 q 必须为非负数。

p越大，长程吸引力越强；而 q越大，长程排斥力越弱。在每个时间步，合力都会移动节点，直到达到收敛。

布局质量主要受 p 和 q影响。分别对 p 和 q使用指数 $(2,1)$ 时，FR模型 [1]会生成边长均匀的布局。LinLog模型 [15] 使用指数$(0,1)$，以便更好地揭示图中的簇。最近，Jacomy等人 [16]建议使用指数$(1,1)$，作为这两种模型之间的中间模型；许多可视化库或系统（如GraphViz [32] 或D 3 [9]）也将这些值设为默认值。

缺点。吸引弹簧力和电学排斥力能够满足小型简单图的**P1** 和 **P2** 。然而，对于大型图，基于幂函数的此类力可能无法充分表示图邻域和聚类结构。接下来，我们将详细说明经典FDP的缺点，具体分析图 2中展示的两个问题案例。

首先，如上所述，当两个节点之间的欧几里得距离趋近于零时，排斥力往往会变得极大。因此，紧密相连的节点可能会彼此远离（见图 2(a)中的 V_1 ）。这样一来，得到的欧几里得距离往往与图论距离不相关（见图1(a)）。这不利于图探索。

其次，过强的长程吸引力可能会使一个簇分裂成多个较小的部分。如图 2(b) 所示，节点 V_2 受到来自一个离群点的强长程吸引力影响，因而移出了主簇。减弱长程吸引力或增强短程吸引力或许能缓解这一问题。然而，调整弹簧-电力模型中的相应系数，总会同时改变长程力和短程力。

尽管如多层次方法 [7]，[33]，[34],中那样，为不同层级的力赋予不同系数可以缓解上述两个问题，但基于幂函数的力本质上难以避免在短程范围内产生极大的排斥力和极小的吸引力。另一方面，长程范围内较小的排斥力和较大的吸引力有助于减少长边，从而形成具有合理全局结构的布局。因此，

我们提出了新模型，将这些效应解耦，使短程范围内的排斥力和吸引力能够得到改善，同时在长程范围内保留传统 FDP 模型的特性。

3.2 t 力

避免排斥力变得极大的一个直接方法，是在两个节点彼此靠近时将力值设为常数。然而，仅将排斥力设为常数无法满足 **P3**，因为很难找到与弹簧式吸引力之间的良好平衡。因此，较好的做法是在短距离内增强吸引力并减小排斥力，使邻居能够彼此靠得更近。为此，我们提出 t 力作为一种新的短程力，发挥两种作用：一种新的排斥力，以及短距离内吸引力的一个组成部分。

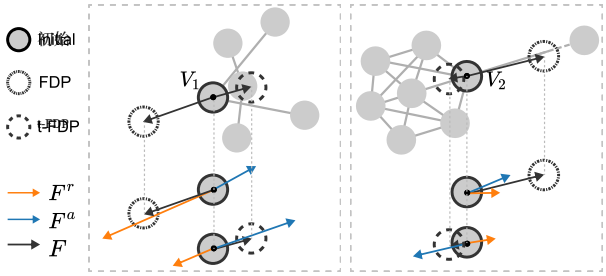


图 2. 传统 FDP 中的两个问题案例：(a) 三个彼此不相邻且距离很近的节点产生很大的排斥力，将 V_1 推出其 cluster；(b) 强大的长程吸引力将 V_2 拉出一个局部簇。t-FDP 避免了这两个问题。

作为一种排斥力，它在本质上应是短程的，因此我们要求它在远距离时较弱，与基于幂函数的排斥力相似。此外，它在短距离内应有上界，以避免接触力过大并提高优化稳定性。最后，为了将其用作短距离内吸引力的附加组成部分，我们进一步要求它具有与线性弹簧力相似的行为；如 Eades [5] 所示，与其他力相比，线性弹簧力相当强。设两个图节点之间的欧几里得距离为 $d = \|\mathbf{x}_i - \mathbf{x}_j\|$，则力 $f(d)$ 应满足以下三个要求：

- **R1:** $\exists \varsigma > 0$ s.t. $0 < f(d) \leq \varsigma$, $\forall d > 0$;
- **R2:** $f(d) \sim d^{-q}$ as $d \to \infty$, where $q > 0$; and
- **R3:** $f(d) \sim d$ as $d \to 0$.

换言之，这种力在两个节点之间距离较短时具有上限 ς，并会随着距离增大而平滑地衰减至零，且衰减速率可以不同。

考察了多种函数后，我们发现以下基于 t 分布的力满足我们的要求。这并不意味着不存在其他可行的函数，但这个函数似乎非常简单，并具备所有必需的特性：

$$f(d) = \frac{2\tau\tilde{\varphi}d}{(1+\tau d^2)^{\tilde{\varphi}+1}}, \tag{3}$$

其中，τ和 $\tilde{\varphi}$ 是不小于零的常数。为简便起见，我们令 τ 为 1，并将式 3 的记法简化为以下函数：

$$f(d) = \frac{d}{(1+d^2)^{\varphi}} \tag{4}$$

指数为 $\varphi \geq 1$。

当 $|d|$ 接近零时，函数 $f(d)$ 趋近于 1；随着 $|d|$ 增大，它逐渐减小至零。我们将式 4 定义的重尾力称为*t* 力。图 3(a) 展示了 $f(d)$ 在 φ 取三个不同值时的情况。其最大值始终小于 1，且位于 d 范围内。 [0,1],$f(d)$ 受 φ 的影响很大。φ 越小，最大值越大，尾部越重。无论 φ 取何值，该函数都有上界 ς。

3.3 t-FDP 模型

使用 t 力，我们现在将基于力的图布局中的排斥力和吸引力定义为：

$$F^r(i,j) = -\frac{\|\mathbf{x}_i - \mathbf{x}_j\|}{(1+\|\mathbf{x}_i - \mathbf{x}_j\|^2)^{\gamma}} \frac{\mathbf{x}_i - \mathbf{x}_j}{\|\mathbf{x}_i - \mathbf{x}_j\|}, \quad i \neq j, \tag{5}$$

$$F^a(i,j) = \left(\|\mathbf{x}_i - \mathbf{x}_j\| + \frac{\beta\|\mathbf{x}_i - \mathbf{x}_j\|}{1+\|\mathbf{x}_i - \mathbf{x}_j\|^2}\right) \frac{\mathbf{x}_i - \mathbf{x}_j}{\|\mathbf{x}_i - \mathbf{x}_j\|}, \quad i \leftrightarrow j, \tag{6}$$

其中，β 是吸引型 t 力的权重。遵循 ForceAtlas2 [16], 的建议，我们使用线性吸引弹簧力，以及满足 $\varphi = 1$的吸引型 t 力。

作用于第 i 个节点的两种力合并后，得到以下合力：

$$F(i) = \sum_{i \neq j} -\frac{||\mathbf{x}_i - \mathbf{x}_j||}{(1 + ||\mathbf{x}_i - \mathbf{x}_j||^2)^{\gamma}} \frac{\mathbf{x}_i - \mathbf{x}_j}{||\mathbf{x}_i - \mathbf{x}_j||} + \alpha \sum_{i \leftrightarrow j} \left(||\mathbf{x}_i - \mathbf{x}_j|| + \frac{\beta ||\mathbf{x}_i - \mathbf{x}_j||}{1 + ||\mathbf{x}_i - \mathbf{x}_j||^2} \right) \frac{\mathbf{x}_i - \mathbf{x}_j}{||\mathbf{x}_i - \mathbf{x}_j||}, \quad (7)$$

其中，γ 是排斥型 t 力分量的指数，α 是吸引力的权重。该模型保留了原模型中的弹簧吸引力，同时加入短程吸引型 t 力，并将其排斥力替换为短程排斥型 t 力。短程吸引型 t 力的形式与 tsNET 中的类似（参见补充材料），并由一个根据图论距离推导出的额外项加权。与 tsNET 不同，我们的短程吸引型 t 力仅作用于给定的边，并与长程弹簧力结合，以更好地保持全局结构（见第 4.2 节）。

参数分析。参数 γ 规定了排斥型 t 力的范围和大小，用以控制布局中邻居之间的最大距离；而 α 和 β则分别调整长程和短程吸引型 t 力的权重。为满足上述三个原则，必须合理设置这些参数。然而，对于大型图，分析它们之间的关系并不容易，因为每个节点都会受到许多其他节点产生的潜在力的影响。为简化分析，我们首先从两个相连节点这一极端情况出发，考察这些参数的有效范围，然后为一般图提供指导建议。

为指定 α 和 β，我们将 **P2** 形式化表示为两个相连节点 i 和 j之间排斥力与吸引力的关系：

$$\lim_{|\mathbf{x}_i - \mathbf{x}_j| \to 0} \frac{F^a(i,j)}{F^r(i,j)} < 1. \quad (8)$$

将公式 5 和公式 6 代入公式 8，可得：

$$\alpha(1 + \beta) < 1. \quad (9)$$

可以看出，α 和 β 不能同时很大，也不能同时很小，否则无法平衡吸引力和排斥力。 α越小，长程吸引力越弱，越有可能产生长边，从而形成节点均匀分布的布局（见使用 $\alpha = 0.01$的图 4(a)）。反之，长边会增多，布局趋于分裂成许多子簇（见采用 $\alpha = 0.50$的图 4(a)）。增大 β 会增强短程吸引力，使节点更有可能聚集

在一起，形成更多子簇（见采用 β =50的图 4(b)）。对于固定的 β，较小的 α 会使 $\alpha\beta$减小，并提高排斥力所占比例。在这种情况下，布局倾向于使节点分布得更加均匀（见使用 $\alpha = 0.01$的图 4(a)）。可以看出，图 4(a) 和图 4(b) 中呈现出类似的趋势。

对于固定的 $\alpha\beta$，减小 α 会提高短程吸引力和排斥力相对于总体力的比例，从而更好地保持图

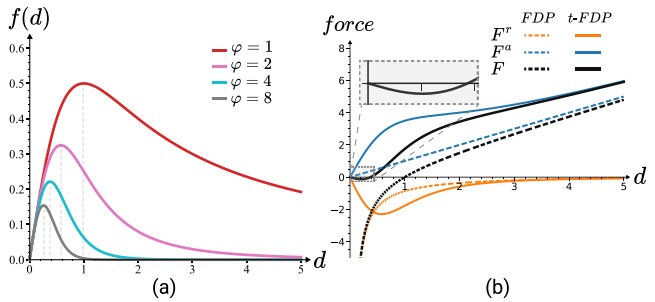


图 3。(a) 不同 φ 下 t 力的函数图；(b) FDP 与 t-FDP 之间作用于两个相连节点的排斥力（橙色）、吸引力（蓝色）和合力（黑色）的比较，分别以虚线和实线表示。主要区别在于短程力，在 t-FDP 中该力是有界的。

邻域（参见图 4(b) 左侧的一环邻域保持程度）。然而，距离保持效果会变差（参见图 4(b) 中的应力误差），因为较弱的长程吸引力会导致出现许多长边。

为指定 γ，我们正式表述 **P3**，该条件要求两个相连节点之间的吸引力大于排斥力，以形成可靠聚类，但节点彼此接近时除外。因此，F^a 和 F^r 满足以下条件：

$$F^a(i,j) > F^r(i,j) \quad \text{if} \quad \|\mathbf{x}_i - \mathbf{x}_j\| > \epsilon$$

其中，ϵ 是一个大于零的小值。由于 t 力是一种短程力，在较长距离下，吸引力由于其弹簧力分量而必然大于排斥力。然而，对于较短距离，该条件可能不成立。如图 3(a) 所示，指数 γ 越大，力的大小越小。为避免由 $\alpha\beta$ 加权的吸引型 t 力小于排斥力，我们要求：

$$\gamma > 1. \quad (10)$$

如图 3(a) 所示，较大的指数 γ 会使排斥力的作用范围变短。相对而言，合力会在距离较近时产生吸引作用，从而更好地呈现局部聚类结构（见图 5(a)），但无法有效保留邻域和距离（图 5(b)）。

总体而言，γ 指定了控制布局中邻居最远距离的排斥型 t 力的范围和大小，而 α 和 β 分别调整吸引型长程和短程 t 力的权重。在我们的实验中，我们发现，采用 α = 0.1、β = 8 和 γ = 2 的配置效果良好，能够保留局部和全局结构。图 3(b)比较了标准 FDP 和我们的 t-FDP 模型作用于节点上的吸引力、排斥力和合力。当两个节点之间的距离小于某个值时，排斥力大于吸引力；否则，吸引力更大。这与三个要求**R1-R3**十分吻合。

3.4 排斥力的近似计算

计算图中的排斥力，需要计算每个节点与所有其他节点之间的相互作用，计算量为 $O(n^2)$ （我们称之为精确 方法）。正如 [29] 针对 t-SNE 提出的那样，可以采用基于树的

近似方法（如 Barnes–Hut算法（BH）） [18] 来截断 t 力。不过，这仍需要 $O(n\log n)$次计算，对于大型图而言并不可行。随机顶点采样 [20] 的运行时间为 $O(n)$ ，但对于大型图，它生成的布局通常不如其他方法生成的布局。为了解决这一问题，我们提出了一种基于插值的快速傅里叶变换（ibFFT）算法，该算法最初是为加速 t-SNE 而设计的 [12]。它生成的布局与 BH 方法相近，但对于大多数大型图，速度快约十倍。

为此，我们首先将排斥型 t 力表示为多个加权核函数的乘积和：

$$\begin{aligned} F^r(i) &= \sum_{j=1,j\neq i}^{n} \frac{||\mathbf{x}_i-\mathbf{x}_j||}{(1+||\mathbf{x}_i-\mathbf{x}_j||^2)^\gamma}\frac{\mathbf{x}_i-\mathbf{x}_j}{||\mathbf{x}_i-\mathbf{x}_j||} \\ &= \sum_{j=1,j\neq i}^{n} \frac{\mathbf{x}_i-\mathbf{x}_j}{(1+||\mathbf{x}_i-\mathbf{x}_j||^2)^\gamma} \\ &= \mathbf{x}_i\sum_{j=1}^{n}\mathbf{K}(\mathbf{x}_i,\mathbf{x}_j)-\sum_{j=1}^{n}\mathbf{K}(\mathbf{x}_i,\mathbf{x}_j)\mathbf{x}_j \end{aligned} \tag{11}$$

其中，核 $\mathbf{K}(\mathbf{x}_i,\mathbf{x}_j)$ 为：

$$\mathbf{K}(\mathbf{x}_i,\mathbf{x}_j)=\frac{1}{(1+||\mathbf{x}_i-\mathbf{x}_j||^2)^\gamma}. \tag{12}$$

由于$\mathbf{x}_j$ 是一个二维点，因此可以分别计算每个维度：

$$F^r(i)_{(1)}=\mathbf{x}_{i(1)}\sum_{j=1}^{n}\mathbf{K}(\mathbf{x}_i,\mathbf{x}_j)-\sum_{j=1}^{n}\mathbf{K}(\mathbf{x}_i,\mathbf{x}_j)\mathbf{x}_{j(1)} \tag{13}$$

$$F^r(i)_{(2)}=\mathbf{x}_{i(2)}\sum_{j=1}^{n}\mathbf{K}(\mathbf{x}_i,\mathbf{x}_j)-\sum_{j=1}^{n}\mathbf{K}(\mathbf{x}_i,\mathbf{x}_j)\mathbf{x}_{j(2)}, \tag{14}$$

其中，$\mathbf{x}^{j(1)}$ 和$\mathbf{x}^{j(2)}$ 分别是$\mathbf{x}j$的 x坐标和 y坐标。因此， $F^r(i)$ 由核 $\mathbf{K}(\mathbf{x}_i,\mathbf{x}j)$ 与权重 v_j 之间的三组乘积和决定：

$$\psi(\mathbf{x}_i)=\sum_{j=1}^{n}\mathbf{K}(\mathbf{x}_i,\mathbf{x}_j)v_j. \tag{15}$$

其中， v_j 是 1、$\mathbf{x}^{j(1)}$或$\mathbf{x}^{j(2)}$。例如， $F^r(i)^{(1)}$ 通过公式 13 中的两组乘积和计算得出，其中权重v_j 分别为 1 和$\mathbf{x}^{j(1)}$。用朴素方法计算这样的乘积和需要 n^2 时间；当节点数量 n很大时，这种方法的开销难以承受。

或者，Linderman 等人 [12] 利用核$\mathbf{K}$的低秩特性，以少量（$k\times k$个）等间距二维网格节点近似公式 15。该方法分为三个步骤：

- 使用拉格朗日多项式将所有数据点$\mathbf{x}_i$ 投影到网格上，时间复杂度为 $O(k^2n)$;
- 计算网格节点之间的相互作用，该过程可通过快速傅里叶变换加速，复杂度为 $O(2k^2\log k)$；以及
- 将所有网格节点之间的相互作用反投影到原始数据点，时间复杂度为 $O(k^2n)$。

我们可以看到，总体时间复杂度为 $O(k^2n)$ ，其中k 是一个较小的数。

然而，上述多项式插值方案受到龙格现象的影响 [35] ，导致

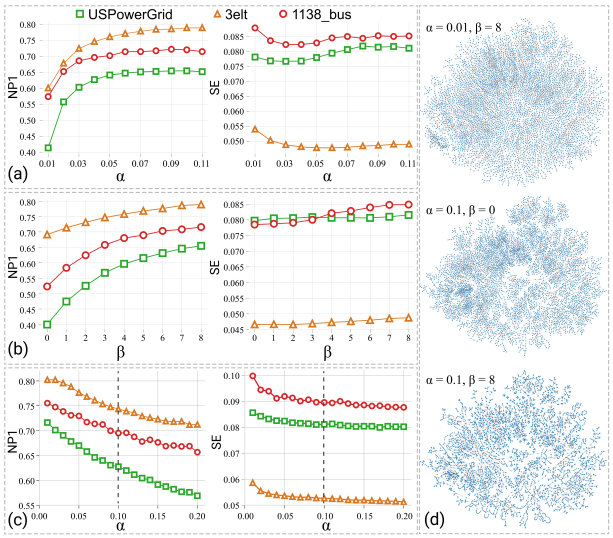


图 4. α 和 β 对图布局的影响：(a) 对于固定的 $\beta=8$，较小的 α 会使 *USpowerGrid* 图的布局中节点分布得更加均匀；(b) 对于固定的 $\alpha=0.1$，较大的 β 会产生更多子簇；(c) 对于给定的三幅图， $\alpha\beta=0.8$ 采用不同的 α 和 β 组合会产生不同的一环邻域保持值（越高越好）和应力误差（越低越好）。 $\alpha=0.1$ 在优化这两个方面之间实现了良好平衡；(d) *USpowerGrid* 图采用不同的 α 和 β组合生成的三种布局结果。

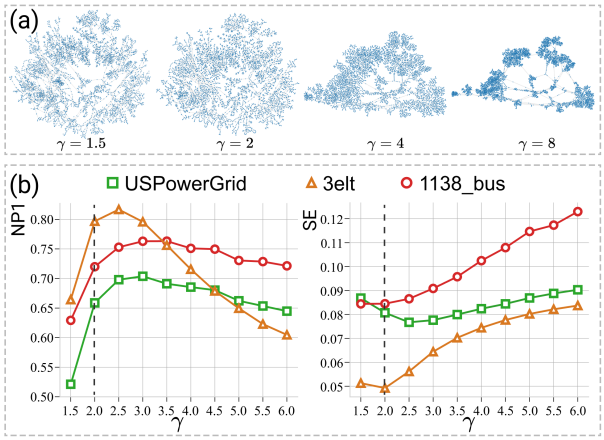


图 5. γ 对图布局的影响：（a） $\gamma=2$，4，8时*USpowerGrid*图的布局；（b）三个图在不同 γ 下的一环邻域保持（左）和应力误差（右）。$\gamma=2$ 在优化这两个方面之间实现了良好的平衡。

其准确性随 k 增大而降低。为解决这一问题，Linderman 等人 [12] 提出将整个布局空间 $[\mathbf{x}_{\min},\ \mathbf{x}_{\max}]\times[\mathbf{x}_{\min},\ \mathbf{x}_{\max}]$ 划分为一系列大小相等的区间，然后在每个区间内使用 $k\times k$ 等间距节点进行多项式插值。具体而言，首先将每个数据点 $\mathbf{x}_i$ 投影到对应的网格方块中，然后计算每个网格方块内 $k\times k$ 等间距网格节点之间的相互作用；最后，将每个网格方块内所有网格节点的相互作用反向投影回原始点。第一步和最后一步仍需要 $O(k^2n)$ 次运算，而借助快速傅里叶变换加速，第二步可在 $O(2(N_{\text{int}}k)^2log(N_{\text{int}}k))$ 次运算内完成。因此，总计算复杂度变为 $O(nk^2+(N_{\text{int}}k)^2log(N_{\text{int}}k))$。

选择 k。Linderman 等人 [12] 建议将 $k=3$

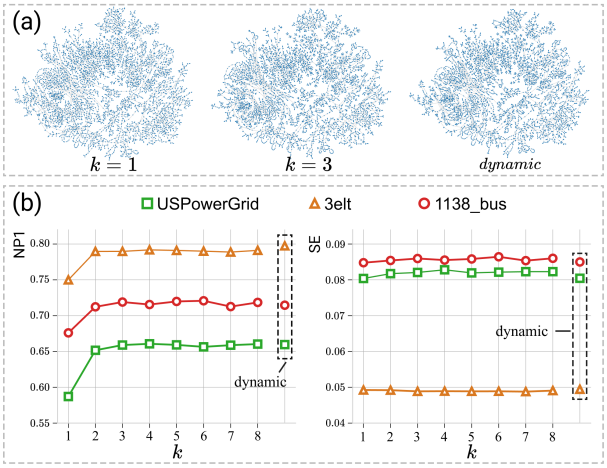


图 6. 不同 k 值对布局的影响。(a) 使用 $k=1$、$k=3$和我们的动态策略生成的*USPowerGrid* 图的布局；(b) 三个图上的一环邻域保持（左）和应力误差（右）随 k 的变化。我们的动态策略的值位于虚线框中。

和 $N_{\text{int}} = \max(50, [y_{\max} - y_{\min}])$ 用于 t-SNE 可视化高维数据。我们发现，使用 $k=3$ 生成的布局与使用$k=1$ 展示全局结构时生成的布局相似（见图 6(a)），但邻域保持效果更好（图 6(b) 左侧）。然而，计算复杂度会随着 k呈二次增长。

为了取得良好的权衡，我们建议在优化过程中改变 $k=1$、2、3 的值：首先在 90%的迭代次数中使用 $k=1$，以获得合理的布局；然后在 5%的迭代次数中使用 $k=2$；最后在剩余的迭代次数中使用 $k=3$。图 6(a)右侧展示了一个示例，其结构几乎与直接使用 $k=3$ 生成的布局相同（见图 6(a)中部）。我们测试了多个数据集，发现这种对 k 值的动态改变能够以最少的计算时间生成高质量的布局（见图 6(b)中的动态策略值）。动态策略的计算时间与设置 $k=1$ 的计算时间接近。

基于对 k 的此类选择，我们的 t-FDP 可以在 O(n) 时间内运行，尤其适用于 N_{int} 远小于 n 的大型图。

4 评估

遵循 UMAP [27]，我们使用 *Python3* 实现了 t-FDP，并使用 *numba* 库 [36] 编译器将 Python 代码编译为快速机器码，其运行速度与纯 C 代码相近。为了利用图形处理器加速，我们使用开源矩阵库 *cupy*，借助 NVIDIA CUDA 加速矩阵计算。为满足各类用户的需求，我们还提供了 JavaScript 实现，可作为“d3-force”库的即插即用式力函数；如果有图形处理器，还可以使用 WebGL 加速。下文将 t-FDP 的基于插值的快速傅里叶变换（ibFFT）实现称为*ibFFT*。

为展示其有效性，我们从两个方面评估 t-FDP。首先，通过将基于 FFT 的近似与精确方法及其他近似方法进行比较，验证我们的基于 FFT 的近似。其次，在一组不同规模的合成图和真实世界图上，将其与几种最先进的布局方法进行比较

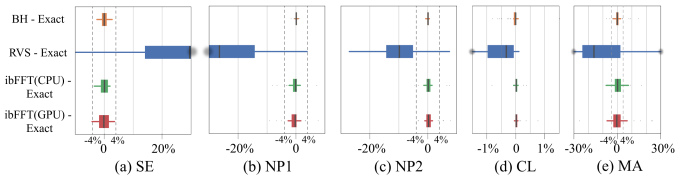


图 7. 比较四种近似方法与 t-FDP 模型的精确方法所生成的布局时，得到的五种相对误差指标（缩写见正文）。

。所有结果均在一台配备 Intel i7-8700 处理器、32 GB CPU 内存和 NVIDIA GeForce RTX 2080 图形处理器（8GB 图形处理器内存）的台式机上测得。

数据集。为进行全面评估，我们收集了 50 个节点数和边数各不相同的图（见补充材料中的表1）。其中大多数选自 Florida Collection [37]，、SNAP 网络集合 [38]，以及 tsNET [2] 和 DRGraph [10] 使用的数据集合。对于 FDP 方法，我们的 t-FDP 模型能够天然地处理多连通分量图，因此纳入了 7 个包含多个连通分量的图。此外，我们使用 graph-tool Python 库 [39] 合成了三个具有不同聚类结构的图。

方法。比较涵盖了十种图布局方法：Fruchterman-Reingold 力导向布局（FR） [1]，、随机顶点采样力导向布局（FR-RVS） [20]，、SFDP [7]，、ForceAtlas2（FA2） [16]，、Linlog [15]，、PivotMDS（PMDS） [40]，、应力模型（SM） [41]，、Maxent 应力模型（Maxent） [6]，、tsNET [2] 和 DRGraph [10]。将我们的方法也纳入后，可将所有方法分为三类：基于力的方法（FR、SFDP、FA2、LinLog 和 t-FDP）、基于应力的方法（PMDS、SM、Maxent）以及邻域嵌入方法（tsNET 和 DRGraph）。我们使用公开的 C++ 库 OGDF [42] 和 GraphViz [32] 执行 FR、SM 和 SFDP，并采用 FA2、Linlog、Maxent、tsNET 和 DRGraph 原论文作者提供的实现。我们自行实现了 PMDS，而未采用 OGDF 提供的方法，因为该方法不稳定，且无法高效处理大型数据。tsNET 使用基于 GPU 的 t-SNE 实现 [43] 进行了加速。对于 FR-RVS，公开的 JavaScript 实现无法处理大型图，因此我们使用 Python 重新实现，并利用 Numba [36] 库进行加速。每种方法均使用原论文或实现提供的默认参数运行。例如，我们将 tsNET 的困惑度设为 40，并为 DRGraph 选择一阶最近邻域。

为保证比较公平，除 SFDP 和 DRGraph 外，所有方法均使用相同的 PMDS 布局进行初始化；SFDP 和 DRGraph 则采用多层方案进行初始化。由于 SFDP 和 DRGraph 本身具有随机性，我们通过对每个方法在一张图上的 5 次运行结果取平均，评估其运行过程中的平均质量。我们在补充材料中提供了随机初始化的结果；结果表明，使用 PMDS 布局进行初始化可以显著改善大多数方法的效果，这与先前在 t-SNE 和 UMAP 中的发现一致 [44]。

指标。遵循 tsNET [2] 和 DRGraph [10], 的评估方法，我们使用四项指标评估图布局：归一化应力误差、邻域保持度、无交叉度和最小角度。前两项指标描述图结构在保持距离和邻域方面的表现，后两项则从边交叉和最小角度的角度反映图可读性。

- 归一化应力误差 (标准误) [6] 为评估布局对距离的保持程度，我们采用如下定义的归一化应力误差：

$$SE = \frac{2}{n(n-1)} \sum_{i<j} \frac{(\bar{s}\|\mathbf{x}_i - \mathbf{x}_j\| - d_{ij})^2}{{d_{ij}}^2},$$

其中，d_{ij} 表示节点i 与 j 之间的最短路径距离，$\bar{s}$是用于对布局进行统一缩放的缩放因子。当节点 i 与 j 在具有多个连通分量的图中不可达时，我们遵循 DRGraph [10] 的做法，将 d_{ij} 定义为无穷大，并且只计算距离有限的节点对的应力误差。为确保公平比较，我们通过最小化归一化应力误差来确定最优缩放因子 $\bar{s}$。更多详情请参阅 Gansner 等人 [6]。

- 邻域保持度（*NP*）。为衡量布局保持邻域的质量，邻域保持度定义为输入图与布局中定义的 k_i近邻图之间的杰卡德相似度：

$$NP = \frac{1}{n} \sum_i \frac{N_G(i,r) \cap N_L(\mathbf{x}_i, k_i)}{N_G(i,r) \cup N_L(\mathbf{x}_i, k_i)},$$

其中，$N_G(i,r)$ 表示图空间中节点 i 的 r环邻居，k_i 是 $|N_G(i,r)|$，而 $N_L(\mathbf{x}_i, k_i)$ 是布局空间中与$\mathbf{x}_i$ 的 k_i 近邻相对应的节点集合。在本研究中，我们计算 1 环和 2 环图邻域的邻域保持度，并分别记为*NP1* 和*NP2*。

- 无交叉度（*CL*）。为表示边交叉的程度，定义无交叉度指标 [45]，将边交叉数归一化：

$$CL = \begin{cases} 1 - \sqrt{c/c_{max}}, & \text{if } c_{max} > 0 \\ 1, & \text{otherwise.} \end{cases}$$

其中，c 表示边交叉数，c_{max} 表示每个图中该数量的理论上界。值越大，表示布局中边交叉越少，布局越好。

- 最小角度（*MA*）。最小角度指标 [45]计算实际最小角度与边之间理想最小角度的平均偏差：

$$MA = 1 - \frac{1}{n} \sum_i^n \frac{|\theta(i) - \theta_{min}(i)|}{\theta(i)}, \quad \theta(i) = \frac{2\pi}{\text{degree}(i)}$$

其中，$\theta_{min}(i)$ 表示节点 i处的实际最小角度。MA 的取值范围为 [0,1]；当所有节点的所有关联边之间的角度均相等时，该值达到最大。

为了比较不同方法，我们计算了由

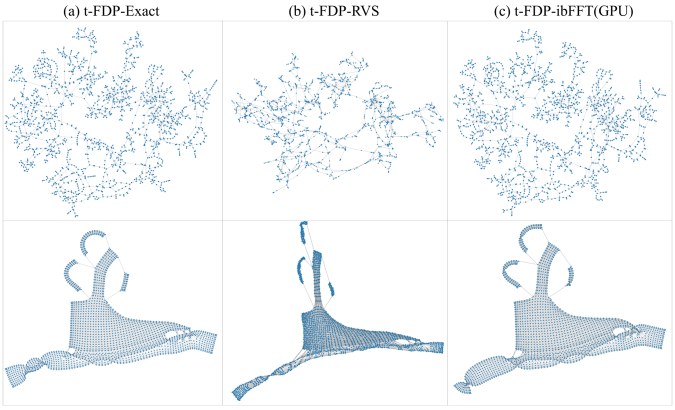


图 8. 精确 t-FDP 模型（a）以及两种近似方法：t-FDP-RVS（b）和 t-FDP-ibFFT(GPU)（c）生成的 *bcspwr07* 和 *walshaw-1* 的可视化结果。

源方法 s 和目标方法 t生成的布局在上述指标上的相对值。以 SE 指标为例，同一图的两个布局$\mathbf{X}_s$ 和$\mathbf{X}_t$ 之间的相对误差为：

$$\overline{SE} = \frac{SE(\mathbf{X}_s) - SE(\mathbf{X}_t)}{SE(\mathbf{X}_t)},$$

对于任何大于零的 $SE(\mathbf{X}_t)$，由于应力误差越小表示距离保持效果越好，因此负的相对误差意味着源方法表现更好。对于其他指标，正的相对误差则意味着源方法表现更好。

4.1 近似方法比较

本实验旨在测量我们 ibFFT 算法的中央处理器版本和图形处理器版本的效率。具体做法是根据质量统计、收敛速率和运行时间表现，将四种近似方法（BH [18], RVS [20], 以及我们 ibFFT 算法的中央处理器和图形处理器版本）与精确方法进行比较。为简便起见，我们分别将这五种方法称为 *t-FDP-BH*、*t-FDP-RVS*、*t-FDP-ibFFT(CPU)*、*t-FDP-ibFFT(GPU)*和 t-FDP-exact。

质量统计。 图 7 中的箱线图总结了四组对应方法在五种布局质量指标上的相对值。

除与 t-FDP-RVS 的比较外，我们可以看到，所有指标的相对值都在 [-4%、4%] 的范围内，且均值非常接近零。与 t-FDP-exact 相比，t-FDP-BH 的四分位距最小，其次是我们方法的两个版本，而 t-FDP-RVS 表现最差。我们的方法在 SE 上产生的相对值略为负值，在 NP2 和 CL 上产生的相对值略为正值，这表明对于某些图，其生成的布局与精确方法得到的布局相似，甚至略好。相比之下，t-FDP-RVS 在 SE 上产生的相对值约为 30%，在 NP1 上产生的相对值则小于 -20%，这表明其生成的布局比精确方法得到的布局差得多。我们推测，NP1 值偏低是由于我们的短程排斥力造成的：RVS 随机抽样得到的部分排斥力可能不足以排斥非邻域节点。请注意

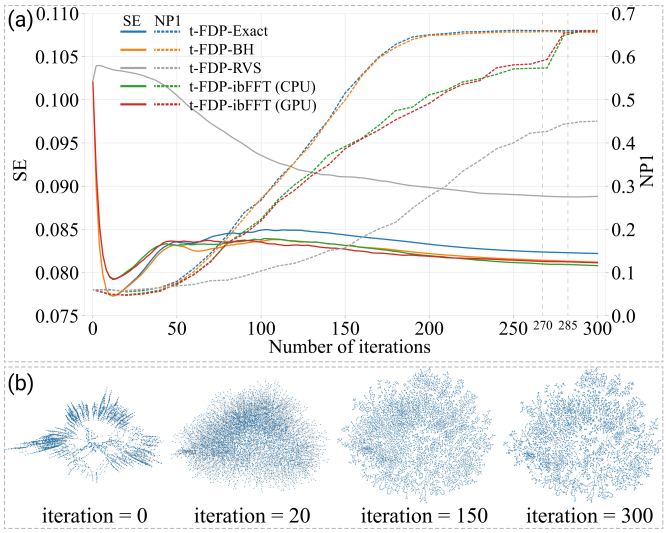


图9. (a) 将 t-FDP 模型的精确方法和四种不同近似方法应用于 *USPowerGrid* 图时，应力误差（标准误，实线）和邻域保持度（NP1，虚线）的收敛情况；(b) 在四个不同迭代次数下生成的布局结果。

t-FDP-ibFFT 的两个版本之间的差异是由 GPU 上的浮点有限精度造成的。

图8展示了将 t-FDP-精确方法、t-FDP-RVS 和 t-FDP-ibFFT（图形处理器）应用于两个图后生成的结果。t-FDP-ibFFT（图形处理器）生成的结果与精确方法几乎相同，在揭示这两个图的树状和网格状结构方面优于 t-FDP-RVS。我们推测，使用随机采样节点计算有界短程排斥力，可能不足以刻画图结构。

收敛速率。为了进一步考察这些方法之间的差异，我们研究了五种不同方法在 USPowerGrid 图上的标准误和 NP1 的收敛速率。由于这些方法每次迭代的运行时间不同，我们考察了随迭代次数变化的收敛情况（见图9），发现所有方法的收敛情况均随迭代次数增加而表现良好。这些方法随时间变化的收敛情况见补充材料。就标准误而言，t-FDP-BH 和 t-FDP-ibFFT 表现相似，且略优于 t-FDP-精确方法。就 NP1 而言，在前270次迭代中，两个版本的 t-FDP-ibFFT 表现都不如其他方法，但随后迅速接近其结果。这是因为前90%的迭代中仅使用一个插值点，而在剩余迭代中使用两个或三个插值点（见第3.4节）。总体而言，我们的方法在邻域保持方面与精确方法相近，但距离保持略优于精确方法及其 BH 近似。我们推测，这是因为近似方法引入的随机噪声可能提升了优化质量 [46]。补充材料展示了这些方法生成的所有结果，其图结构高度相似。相比之下，t-FDP-RVS 的表现与其他方法大不相同，应力误差更大、NP1 值更低，这与图7所示结果一致。

运行时性能。关于运行时性能，

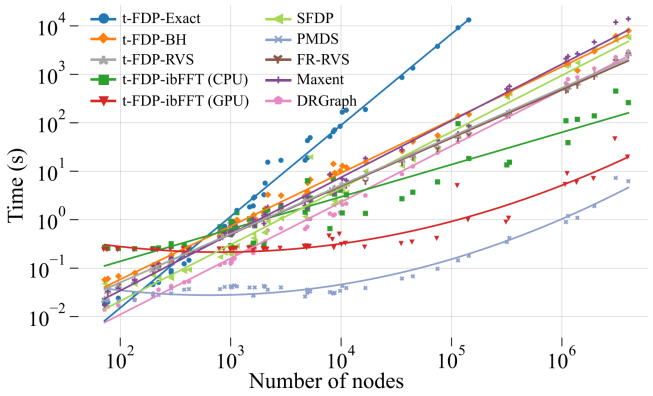


图10. 我们的 t-FDP模型五种不同实现的计算时间，与另外五种能够处理所有数据集的方法进行比较。

图10中的曲线展示了运行时间与图节点数量之间的关系。当节点数量少于500时，t-FDP-ibFFT的两个版本运行时间均为300毫秒，虽然可以接受，但比t-FDP-exact慢。超过这一数量后，ibFFT的优势便显而易见。可以看到，对于大型图，t-FDP-ibFFT（图形处理器）比t-FDP-ibFFT（中央处理器）快一个数量级，也比t-FDP-BH快一个数量级。相比之下，精确方法速度最慢，处理一个包含10万个节点的图需要3个多小时；而我们的GPU版本只需不到10秒，就能为大多数包含数百万个节点的图生成布局。

总体而言，t-FDP-ibFFT（图形处理器）生成的布局与 t-FDP-BH 等方法相似，但处理大型图时速度快得多。因此，我们使用这一实现，在所有数据集上与其他布局方法进行比较。

4.2 布局方法比较

所有方法在各种图上生成的布局截图及完整评分见补充材料。下文将从布局质量统计、可视化结果和运行时性能等方面比较这些布局方法。

质量统计。图11中的热图展示了 11 种不同布局方法针对 50 个图生成的标准误、NP1 和 NP2 值；其他指标见补充材料。每行对应一个图，并按节点数排序；每列对应一种图布局方法。每个单元格显示一个数值，背景颜色表示该行中相应指标的相对值；空白单元格表示该图过大，无法由对应的布局方法处理。可以看到，只有六种方法（FR-RVS、SFDP、PMDS、Maxent、DRGraph 和我们的 t-FDP）能够处理所有图。在大多数数据上，我们的 t-FDP 在应力误差方面表现与 FR 和 SFDP 相近；当节点数少于 500 时，其邻域保持表现与 tsNET 相近。随着节点数增加，它相较于其他方法的优势变得更加明显。

图11(a)显示，Maxent 和 SFDP 对大多数图的效果良好，但在某些图上会产生较大的应力误差（例如，*add32*）；而 tsNET 和 DRGraph 会产生较大的应力

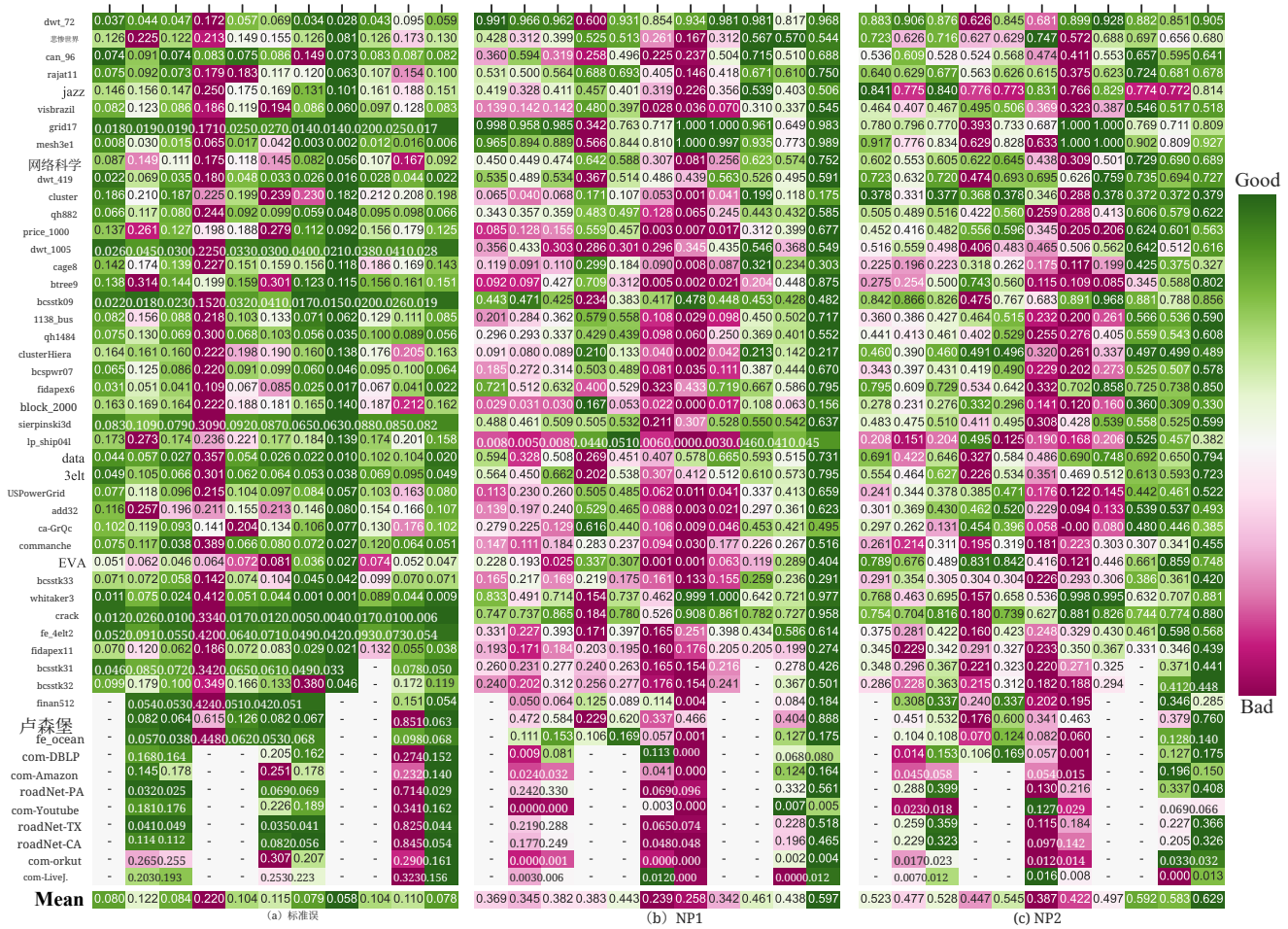


图11. 使用色盲友好粉绿渐变色图绘制热图，展示了 50 个数据集上由十种布局方法生成的布局的 SE（a）、NP1（b）和 NP2（c）值；空白单元格表示图对于相应的布局方法而言过大，无法处理。每行代表一个数据集，每列代表一种布局方法。所有行均根据最佳值和最差值进行相对着色。

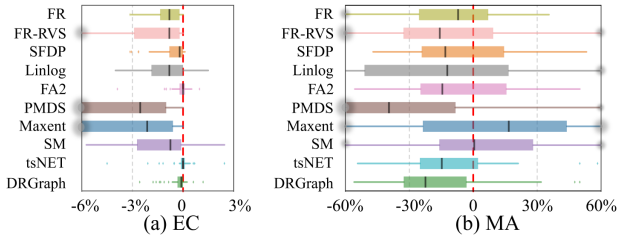


图12. 箱线图汇总了所有布局方法均可处理的 37 个图上的两种相对误差指标 EC 和 MA 的值。两种误差均相对于 t-FDP 计算，值越大越好。

误差，且对大多数图都是如此，甚至在某些示例中会产生最差的布局。我们认为，这两种方法都基于局部邻域，但缺乏用于距离保持的长程吸引力。

图 11(b,c) 显示，对于几乎所有图，我们的 t-FDP 在 NP1 方面表现最佳；对于大多数图，它在 NP2 方面表现最佳或次佳。然而，基于应力的方法（PMDS、Maxent、SM）表现最差，尽管它们在少数网格状图上效果相当不错（例如，*grid17*）。此外，传统的基于力的方法（FR、FR-RVS、SFDP、Linlog 和 FA2）无法为大多数图正确保留邻域。相比之下，基于邻域嵌入的方法（tsNET 和 DRGraph）能有效保留大多数图的邻域，但在应力误差方面，它们的表现仍远逊于 t-FDP。原因是，这些方法旨在形成局部簇，而非捕捉全局结构。值得注意的是，无论采用何种布局算法，一些图的 NP1 和 NP2 值都相当低。图 *lp ship04l* 就是一个极端例子，其 NP1 值为 0.045。仔细检查这些图后，我们发现它们的内在维度很高，几乎不可能实现良好的投影，见图 14(c)。由于 tsNET 中的参数“困惑度”和 DRGraph 中的“k-阶最近邻”对 NP1 和 NP2 有很大影响，我们进一步探究了合适的参数能否带来更好的结果。我们发现，对于大多数数据集，t-FDP 的表现优于 tsNET 和 DRGraph；在测试的一组参数中，其 SE、NP1 和 NP2 的平均值也更高。详细结果见补充材料。

图11底行展示了我们能够使用所有布局方法处理的37个图上三项指标的平均值。SM在标准误方面无疑表现最佳，而我们的t-FDP排名第二。与FR和SFDP相比，t-FDP的标准误表现相近，但在NP1和NP2中分别保留了超过35%和15%的邻域，对应的绝对平均差异分别为0.22和0.10（见图11底行）。尽管Maxent

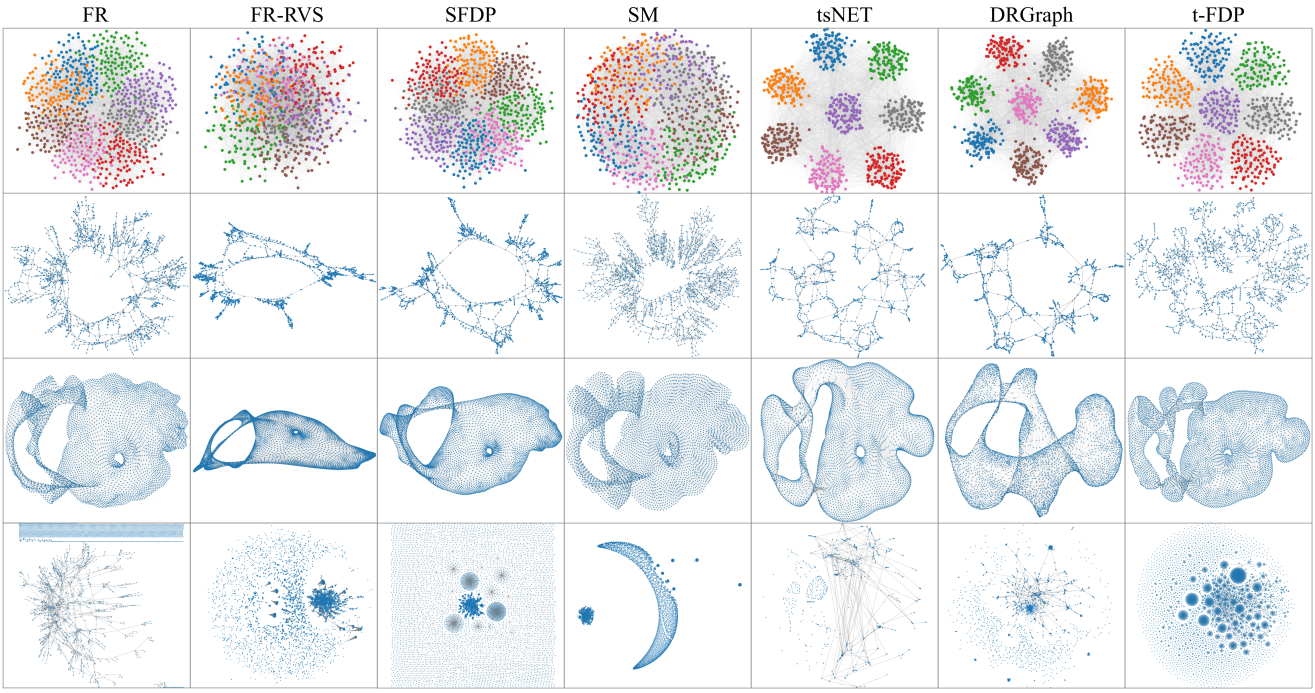


图 13. 四种数据集采用七种方法生成的布局：*cluster*（第一行）、*bcspwr07*（第二行）、*3elt*（第三行）和*eva*（最后一行）。t-FDP 能很好地突出簇，同时在 bcspwr07 中兼顾局部结构与全局结构，并能很好地展开 3elt。

在应力误差方面仅略逊于t-FDP，但在NP1和NP2方面表现最差。同样，PMDS在应力误差、NP1和NP2方面均排名倒数第二。仔细检查Maxent和PMDS生成的布局后，我们发现，它们都无法有效处理边数大于节点数的图。另一方面，SM的应力误差表现最佳、LinLog表现最差，这并不令人意外，因为SM旨在保留距离，而LinLog旨在揭示簇。

图 12 中的箱线图汇总了所有方法均可处理的 37 个图的 EC 和 MA 得分。图 12(a) 显示，t-FDP 在 EC 方面的表现与其他方法相近，而 PMDS 和 Maxent 略逊一筹。图 12(b) 显示，Maxent 的 MA 表现最佳，其次是 SM 和 t-FDP，而 PMDS 表现最差。根据这些结果，我们得出结论：t-FDP 生成的布局可读性与现有方法相当，甚至更胜一筹。

请注意，FR-RVS 在标准误和 NP2 方面显著逊于 FR，这与 Gove [20] 的结果相矛盾。仔细检查结果后，我们发现差异源于初始化：Gove [20]使用的是将节点随机分布在均匀间隔圆盘上的初始化方式。采用类似的随机初始化时，FR-RVS 和 FR 的表现相近，但生成的布局不如采用 PMDS 初始化生成的布局。完整评估见补充材料，其中还包含 FR-RVS 和 t-FDP-RVS 生成的视觉布局以供比较。

可视化结果。 图 13 展示了六种方法应用于四个数据集的可视化结果：*cluster*、*bcspwr07*、*3elt* 和*eva*。对于聚类图（见第一行），可以看到，t-FDP 与 tsNET 和 DRGraph 一样，能够刻画簇，清晰呈现全局结构，同时像 FR 和 SM 一样保留邻域结构（见第二行）；对于网格状结构，能像 FR 一样实现良好的展开效果（见第三行）；并能忠实呈现多分量结构（见最后一行）。尽管 SM 通过保留成对距离维持了全局结构，但其局部结构保留能力和聚类能力最差。FR 和 SFDP 的表现与 SM 相似。即使原始图中不存在聚类结构，tsNET 和 DRGraph 似乎也会形成簇，而且难以维持全局结构。例如，它们将树状图*bcspwr07* 的分支挤压成几个相互重叠的小型子簇（见图 13 第二行）。请注意，我们没有采用任何打包算法来排列多个连通分量，而 FR 和 SFDP 则将它们紧密地打包在布局空间中。

运行时性能。Zhu 等人 [10] 在实验中表明，只有 SFDP、PMDS 和 DRGraph 能处理拥有数百万个节点的图。除了这些方法外，我们还纳入 Maxent，以便全面比较 t-FDP 的运行时性能。对于所有方法，运行时间仅包括布局时间，不包括数据加载和布局初始化等数据处理步骤。图10报告了每个图的运行时间。

与 Zhu 等人的报告相反， [10], 我们的 PMDS 实现速度最快，因为其具有线性计算复杂度。对于节点数少于 1K 的小型图，所有方法都能在一秒内完成处理；而对于较大的图，各方法的性能则有显著差异。对于节点数超过 10万个节点 的图，SFDP、Maxent 和 DRGraph 的性能相近，因为它们计算排斥力时的计算复杂度相似 $O(n\log n)$。基于 ibFFT 的 t-FDP 的 GPU 版本比这些方法快两个数量级，这归功于其 ibFFT近似

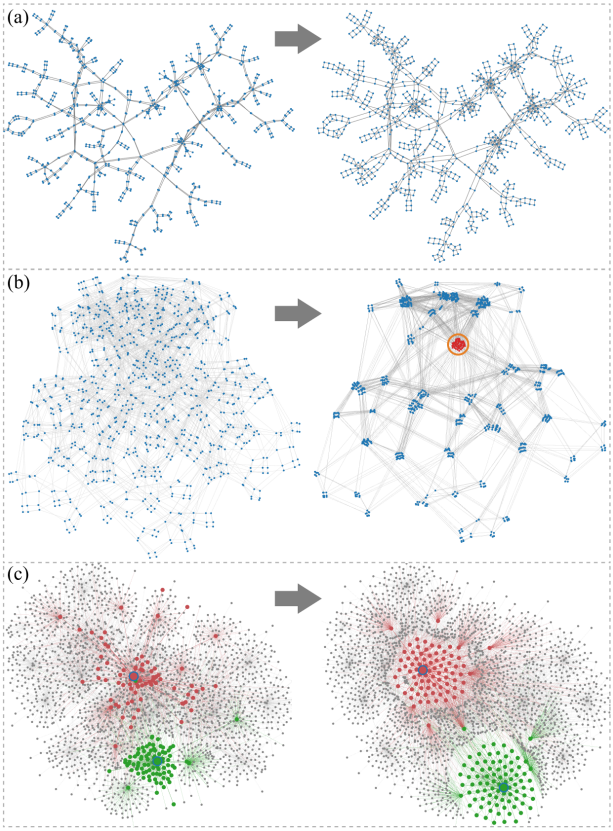


图 14. 通过施加较大的排斥力（a）、施加作用范围较短的排斥力（b），以及局部改变吸引力和排斥力（c）来细化 t-FDP 图布局。由此分别得到分布均匀的局部邻域（a）、主要簇（b）和鱼眼视图（c）。

。例如，GPU版本的 t-FDP 模型处理 *com-LiveJournal* 图（400 万个节点和 3500 万条边）需要 30 秒，而 DRGraph、SFDP 和 Maxent 分别需要 2958 秒、 5745 秒和 11954 秒。

根据五项指标和运行时性能的结果，我们可以得出结论：我们的 t-FDP 模型能够为大多数图生成高质量布局，同时计算速度极快。

5 扩展

由于我们的 t-FDP 模型保留了传统 FDP 的灵活性和简洁性，因此它继承了这些模型的所有可能扩展，例如多层布局方法 [7]或约束布局 [47]。除此之外，我们还可以进一步扩展 t-FDP 模型，通过全局和局部调整排斥型 t 力，更好地支持对图的交互式探索。由于模型速度足够快，这些细化扩展可以通过实时交互实现。

全局细化。 用户通常希望探索不同的图结构，例如详细的局部邻域或骨架状结构。以 t-FDP 布局作为初始化，我们可以通过使用不同大小的排斥型 t 力重新应用 t-FDP 来实现此类可视化。例如，使用较大的排斥型 t 力会使附近的节点均匀分布，从而呈现详细的局部邻域。另一方面，较小的排斥型 t 力会使相连节点彼此靠近，并显现主要结构。

图 14(a) 展示了一个示例：对图 *qh882* 的布局施加较大的排斥力（左图），使分支末端的节点均匀分布（右图），并将 NP1 分数从 0.585 提高到 0.643。在两种结果中，变形效应 [48] ——即外围节点趋于彼此靠近的现象——都得到显著缓解，同时图的所有部分也得以展开。图 14(b) 中的示例则以相反的方式获得：施加作用范围较短的排斥力（较大的 γ），可以清晰地呈现图 *cage8* 的骨架结构。

局部细化。探索过程中，一项主要任务是查找并检查某些节点的邻域。尽管我们的 t-FDP 在邻域保持方面表现良好，但无法确保始终妥善保留所有邻域。图 14(c) 左侧展示了一个 *lp ship04l* 图的示例，其中一些相邻节点（红色和绿色节点）由于各自所属的局部簇而无法恰当放置。为缓解这一问题，我们增强焦点节点与其邻居之间的吸引力，将它们拉到一起，同时施加排斥力以突出显示它们。此外，我们还在其他节点之间施加较大的排斥力，以压缩周围区域。由此生成鱼眼式可视化。图 14(c) 右侧展示了结果，其中红色和绿色节点的大多数邻域都清晰可见。请注意，仍有少数节点未被拉到一起，这是因为它们各自所属的局部簇。

6 结论与未来工作

我们提出了 t-FDP，一种用于图布局的新型 FDP模型。它基于这样的观察：现有的 FDP模型无法正确捕捉局部邻域，尤其是当两个节点重叠时会产生较大的接触力。

因此，我们设计了一种基于 t 分布的新型短程力：t 力。它具有明确的上界，在远距离时的行为类似于基于幂函数的现有吸引力，在短距离时则类似于排斥力。此外，我们采用 t-SNE 中使用的基于快速傅里叶变换的近似策略，以加速排斥力的计算。我们将 t-FDP模型与多种最先进的布局方法进行了定量比较，结果表明，基于快速傅里叶变换近似的 t-FDP 在大多数情况下优于这些方法，并且在中央处理器上快一个数量级，在图形处理器上快两个数量级。最后，我们展示了 t-FDP 在探索不同图结构方面的实用性。

我们的方法仍有一些局限性，未来希望加以解决：首先，t-FDP 在距离保持方面略逊于应力模型，但优于 tsNET 和 DRGraph。因此，我们计划探索改进其长程力的可能性。其次，t-FDP 将用于平衡作用于每个节点的吸引力与排斥力的单个参数替换为三个参数 α, β 和 γ，这可能会给用户带来额外的复杂性。另一方面，这些参数为根据不同任务调整方法提供了灵活性 [3], 因此，我们希望探索自动参数调优方法。最后，我们计划研究其他可能的短程力形式，以进一步提高布局质量，并探索它们在降维中的应用。

致谢

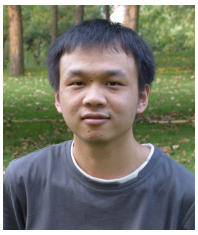
作者感谢匿名审稿人提供的宝贵意见。本研究获得了中国国家重点研发计划（2019YFB1704201）和国家自然科学基金委员会（62132017、62141217）资助，同时也得到德国研究联合会（DFG）在德国卓越战略项目下的资助——EXC 2117 - 422037984。

参考文献

[1] T. M. J. Fruchterman 和 E. M. Reingold，“基于力导向布局的图绘制”，Software: Practice and Experience，第 21 卷，第 11 期，第 1129–1164 页，1991 年。 [在线]。网址：https: //onlinelibrary.wiley.com/doi/abs/ 10 1002 4380211102 . /spe。J. F. Kruiger、P. E. Rauber、R. M. Martins、A. Kerren、S. Kobourov 和 A. C. Telea，“基于 t-SNE 的图布局”，《计算机图形论坛》，第 36 卷，第 3 期，第 283–294 页，2017 年。 [在线]。网址：https://onlinelibrary.wiley.com/doi/abs/ 10 1111 13187 . /cgf。B. Lee、C. Plaisant、C. S. Parr、J.-D. Fekete 和 N. Henry，“图可视化的任务分类体系”，载于《2006 年 AVI 研讨会论文集：超越时间与误差：信息可视化的新型评估方法》，2006 年，第 1–5 页。[4] S. G. Kobourov，《力导向绘图算法》，R. Tamassia 编。牛津：Chapman and Hall/CRC，2013 年。[5] T. Biedl 和 G. Kant，“正交图绘制的一种更优启发式算法”，Computational Geometry，第 9， 3， 159 180, 卷，1998 [ ] 9， 3， 159 180, 期，第 – 页。在线。1998 [ ] 网址：https://www.sciencedirect.com/science/ article/pii/S 0925772197000266 [6] E. R. Gansner、Y. Hu 和 S. North，“用于图布局的最大熵应力模型”，IEEE可视化与计算机图形学汇刊，第 19 卷，第 6 期，第 927–940 页，2013 年 6 月。[7] Y. Hu，“高效、高质量的力导向图绘制”，Mathematica journal，第 10 卷，第 1 期，第 37–71 页，2005年。[8] M. Bastian、S. Heymann 和 M. Jacomy，“Gephi：用于探索和操作网络的开源软件”，载于《AAAI网络与社交媒体国际会议论文集》，第 3 卷，第 1 期，2009 年，第 361–362 页。[9] M. Bostock、V. Ogievetsky 和 J. Heer，“D^3 数据驱动文档”，IEEE可视化与计算机图形学汇刊，第 17 卷，第 12 期，第 2301–2309 页，2011 年 12 月。[10] M. Zhu、W. Chen、Y. Hu、Y. Hou、L. Liu 和 K. Zhang，“DRGraph：一种通过降维实现大规模图高效布局的算法”，IEEE可视化与计算机图形学汇刊，第 27 卷，第 2 期，第 1666–1676 页，2021 年 2 月。[11] L. van der Maaten 和 G. Hinton，“使用 t-SNE 可视化数据”，Journal of Machine Learning Research，第 9 卷，第 86 期，第 2579–2605 页，2008年。 [在线]。网址：http://jmlr.org/papers/v9/ vandermaaten08a.html[12] G. C. Linderman、M. Rachh、J. G. Hoskins、S. Steinerberger 和 Y. Kluger，“用于改进单细胞 RNA 测序数据可视化的快速基于插值的 t-SNE”，Nature Methods，第 16 卷，第 3 期，第 243–245 页，2019 年 3 月。 [在线]。网址：https://doi.org/ 10 1038 41592 018 0308 4 . /s - - - [13] W. T. Tutte，“ 如何绘制图 ”，Proceedings of the London Mathematical Society，第 s3-13 卷，第 1 期，第 743–767 页，1963 年。 [在线]。网址：https://londmathsoc.onlinelibrary.wiley.com/doi/abs/ 10 1112 3 13 1 743 . /plms/s - . . Y. Chen、Z. Guan、R. Zhang、X. Du [14] 和 Y. Wang，“探索图数据中关联关系的可视化方法综述”，Journal of Visualization，第 22 卷，639，2019 3，625期，第 – 页。[15] A. Noack，“用于图聚类的能量模型”，Journal of Graph Algorithms and Applications，第 11 卷，第 2 期，第 453–480 页，2007 年。[16] M. Jacomy、T. Venturini、S. Heymann 和 M. Bastian，“Forceatlas2：一种用于便捷网络可视化的连续图布局算法，专为 Gephi 软件设计”，PLOS ONE，第 9 卷，第 6 期，第 1–12 页，2014 年 6 月。 [在线]。网址：https://doi.org/ 10 1371 0098679 . / [17] journal.pone。P. Kumar 和 K. Zhang，“通过节点交错实现聚类有向无环图的可视化”，载于《2009 年 ACM应用计算研讨会论文集》。美国纽约州纽约：计算机协会，2009 年，第 1800–1805 页。[在线]。网址：https://doi.org/10.1145/1529282.1529685

[18] J. Barnes 和 P. Hut，“一种分层 o(n log n) 力计算算法”，《自然》，第 324 卷，第 6096 期，第 446–449 页，1986年12月。 [ ] 10 1038 324446 0 在线。可访问：[19] https://doi.org/。/a L. F. Greengard，“粒子系统中势场的快速计算”，博士学位论文，美国，1987年，aAI8727216。 [在线]。可访问：https://dl.acm.org/doi/ 10 5555 913529 。/ [20] R. Gove，“ 一种用于图布局的随机采样 o(n) 力计算算法 ”，《计算机图形论坛》，第 38 卷，第 3 期，第 739–751 页，2019年。 [在线]。可访问：https://onlinelibrary.wiley. com/doi/abs/ 10 1111 13724 。/cgf. T. Kamada 和 S. Kawai，“一种绘制一般无向图的算法”，《信息处理快报》，第 31 卷，第 1 期，第 7–15 页，1989年4月。 [ ] 10 1016 0020 0190 89 90102 6 在线。[21] 可访问：https://doi.org/。/-()- Y. Koren 和 A. Civril，“用于图绘制的二元应力模型”，载于《国际图绘制研讨会》。Springer，2008年，第 193–205 页。[22] [23] L. Chen 和 A. Buja，“用于非线性降维、图绘制和邻近性分析的局部多维尺度分析”，《美国统计协会会刊》，第 104 卷，第 485 期，第 209–219 页，2009年3月。 [在线]。可访问：https://doi.org/ 10 1198 2009 0111 。/jasa. . Y. Wang、Y. Wang、[24] Y. Sun、L. Zhu、K. Lu、C.-W. Fu、M. Sedlmair、O. Deussen 和 B. Chen，“重新审视应力主元化：交互式约束图可视化的统一框架”，IEEE可视化与计算机图形学汇刊，第 24 卷，第 1 期，第 489–499 页，2018年。[25] K. Pearson，“ X. 对进化数学理论的贡献——II. 均质材料中的偏斜变异”，《伦敦皇家学会哲学汇刊》(A.)，第 186 卷，第 343–414 页，1895年。[26] J. Tang、J. Liu、M. Zhang 和 Q. Mei，“大规模高维数据的可视化”，载于《第25届国际万维网会议论文集》。日内瓦共和国及州，瑞士：国际万维网会议指导委员会，2016年，第 287–297 页。 [在线]。可访问：https://doi.org/ 10 1145 2872427 2883041 。/ . L. McInnes、J. Healy 和 [27] J. Melville，“UMAP：用于降维的均匀流形近似与投影”，arXiv 预印本 arXiv:1802.03426，2018年。[28] E. Amid 和 M. K. Warmuth，“TriMap：使用三元组进行大规模降维”，arXiv 预印本 arXiv:1910.00204，2019年。 [ ] 10 48550 1910 00204 在线。可 [29] 访问：https://doi.org/。/arXiv. . L. Van Der Maaten，“使用基于树的算法加速 t-SNE”，Journal of Machine Learning Research，第 15 卷，第 1 期，第 3221–3245 页，2014年1月。 [在线]。可访问：https: //dl.acm.org/doi/ 10 5555 2627435 2697068 。/ . N. Pezzotti、J. [30] Thijssen、A. Mordvintsev、T. H¨ollt、B. Van Lew、B. P. Lelieveldt、E. Eisemann 和 A. Vilanova，“通用图形处理器线性复杂度 t-SNE 优化”，IEEE可视化与计算机图形学汇刊，第 26 卷，第 1 期，第 1172–1181 页，2020年1月。[31] J. Douglas Carroll 和 P. Arabie，《多维尺度分析》，丛书：《感知与认知手册》（第二版）。圣迭戈：学术出版社，1998年1月。 [在线]。可访问：https://www. sciencedirect.com/science/article/pii/B 9780120999750500051 [32] J. Ellson、E. Gansner、L. Koutsofios、S. C. North 和 G. Woodhull，“Graphviz——开源图绘制工具”，载于《图绘制》，P. Mutzel、M. J¨unger 和 S. Leipert 编。柏林，海德堡：Springer柏林海德堡，2002年，第 483–484 页。[33] C. Walshaw，“一种用于力导向图绘制的多层次算法”，载于《第8届国际图绘制研讨会论文集》，丛书：GD ’00。柏林，海德堡：Springer-Verlag，2000年，第 171–182 页。 [在线]。可访问：https://dl.acm.org/doi/10.5555/ 647552.729414[34] S. Hachul 和 M. J¨unger，“使用基于势场的多层次算法绘制大型图”，载于《国际图绘制研讨会》。Springer，2004年，第 285–295 页。[35] J. F. Epperson，“关于 Runge 示例”，《美国数学月刊》，第 94 卷，第 4 期，第 329–341 页，2022/03/11/ 1987，完整出版日期：1987年4月。 [在线]。可访问：https://doi.org/ 10 2307 2323093 。/ S. K. Lam、A. Pitrou 和 S. Seibert，“Numba：基于 LLVM 的 Python 即时编译器”，载于《第二届高性能计算LLVM编译器基础设施研讨会论文集》，2015年，第 1–6 页。[36] [37] T. A. Davis 和 Y. Hu，“佛罗里达大学稀疏矩阵集合”，ACM Trans. Math. Softw.，第 38 卷，第 1 期，2011年12月。[在线]。可访问：https://doi.org/10.1145/2049662.2049663

[38] J. Leskovec 和 A. Krevl，“SNAP 数据集：斯坦福大型网络数据集集合”，http://snap.stanford.edu/data，2014年6月。[在线]。可访问：http://snap.stanford.edu/ [39] T. P. Peixoto，“graph-tool Python 库”，2014年。 [在线]。可访问：http://figshare.com/articles/graph tool/1164194 [40] U. Brandes 和 C. Pich，“用于大型数据渐进式多维尺度分析的特征值求解器方法”，载于图绘制，M. Kaufmann 和 D. Wagner 编。柏林，海德堡：Springer柏林海德堡，2007年，第 42–53 页。[41] M. Khoury、Y. Hu、S. Krishnan 和 C. Scheidegger，“通过低秩应力主化绘制大型图”，《计算机图形论坛》，第 31 卷，第 3pt1 期，第 975–984 页，2012年。[在线]。可访问：https://onlinelibrary.wiley.com/doi/abs/ 10.1111/j.1467-8659.2012.03090.x [42] M. Chimani、C. Gutwenger、M. J¨unger、G. W. Klau、K. Klein 和 P. Mutzel，“开放图绘制框架（OGDF）。”《图绘制与可视化手册》，第 2011 卷，第 543–569 页，2013年。[43] D. M. Chan、R. Rao、F. Huang 和 J. F. Canny，“t-SNE-CUDA：图形处理器加速的 t-SNE 及其在现代数据中的应用”，载于第*30*届国际计算机体系结构与高性能计算研讨会，2018年9月，第 330–338 页。[44] D. Kobak 和 G. C. Linderman，“初始化对于在 t-SNE 和 UMAP 中保留全局数据结构至关重要”，《自然·生物技术》，第 39 卷，第 2 期，第 156–157 页，2021年2月。 [在线]。可访问：https://doi.org/10.1038/s41587-020-00809-z [45] H. C. Purchase，“图绘制美学的度量指标”，《可视语言与计算杂志》，第 13 卷，第 5 期，第 501–516 页，2002年10月。 [在线]。可访问：https://www.sciencedirect.com/science/ article/pii/S1045926 02902326 X [46] J. Schneider 和 S. Kirkpatrick，随机优化。Springer Science & Business Media，2006年。[47] T. Dwyer，“可扩展、灵活且简单的约束图布局”，《计算机图形论坛》，第 28 卷，第 3 期，第 991–998 页，2009年。 [在线]。可访问：https://doi.org/10.1111/j.1467-8659.2009.01449.x [48] Y. Hu 和 Y. Koren，“扩展弹簧电模型以克服翘曲效应”，载于*2009*年 *IEEE* 太平洋可视化研讨会，2009年4月，第 129–136 页。



Fahai Zhong 是山东大学计算机科学与技术学院三年级硕士研究生。他的研究兴趣包括图可视化和降维。



Mingliang Xue 是山东大学计算机科学与技术学院的五年级博士生。他的研究兴趣包括图可视化和降维。



Jian Zhang 于2005年获得明尼苏达大学应用数学博士学位。在宾夕法尼亚州立大学完成博士后研究后，他现任中国科学院（CAS）计算机网络信息中心教授。他目前的研究兴趣包括科学计算和科学可视化。



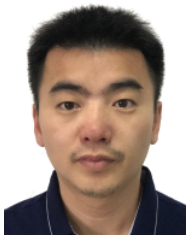
张帆 于2009年和2015年分别获得中国山东大学计算机科学与技术学院工学学士和博士学位。目前，他是中国山东工商学院计算机科学与技术学院副教授。他的研究兴趣包括计算机图形学、计算机视觉和人工智能。



Rui Ban 是中国信息技术设计咨询研究院有限公司智能网络设计研究院的高级工程师。他的研究兴趣包括大数据可视化和视觉分析。



OliverDeussen 毕业于卡尔斯鲁厄理工学院，现任德国康斯坦茨大学教授，并担任中国科学院深圳研究院客座教授。他曾担任《计算机图形论坛》联合主编，并曾任欧洲计算机图形学协会主席。他的研究领域包括复杂生物系统的建模与渲染、非真实感渲染以及信息可视化。



YunhaiWang 是山东大学计算机科学与技术学院教授。他担任《计算机图形论坛》副编辑。他的研究兴趣包括科学可视化、信息可视化和计算机图形学。