

对于Set#1，默认条件为Servers number = 20, Density = 1.0, hops = 2。在Density和hops不变，Servers number由10变化至35，EC-EDP-O算法的计算开销随着边缘节点数目的变化呈现指数增长趋势，虽然该算法能计算出当前条件下最优解，但其时间损耗远大于其他算法。且其他算法在节点数目比较小的情况下，几乎没有时间损耗上的差距。如Fig. 6(a)中，当节点数目由30增长至35时，EC-EDP-O算法的计算开销从1000ms上升至2100ms，增加1.1倍；同时EC-EDP-V算法能以接近1ms的开销达到最优解96%的结果，这足以说明EC-EDP-V算法在小数据集范围内的高效性。Fig. 6(b)和Fig. 6(c)也分别展示出Density和Hops对算法计算开销的影响，Density和Hops升高，图的复杂程度和搜索的范围不断加大，导致EC-EDP-O的计算开销远大于其他算法，这也显示出EC-EDP-O在高密度高复杂度场景中的适用性是极低的。

在Set#2中，我们对EC-EDP-V和GD、Random和Replica等算法在节点数较大的拓扑网络中进行了高效性测试，默认条件变为了Servers number =150, Density = 2.0, hops = 2。在三种不同的测试条件下，4种模型计算开销的增长趋势十分相近。尽管EC-EDP-V的时间开销与其他算法的差距随着拓扑复杂程度的增大有所增加，但差距增长的趋势相对EC-EDP-O来说十分缓慢，并且EC-EDP-V能以合理的额外开销换取相对其他方案40%以上的优化，同时相对于EC-EDP-O来说，在保持十分接近最优解的前提下，大大降低了计算开销，达到了十分不错的Trade-off。说明在大数据集场景下，EC-EDP-V仍然能够保持不错的高效性。

总而言之，EC-EDP-V在各种场景下都有足够的适应性，能够在保持低开销的同时，拥有极其贴近最优解的性能。