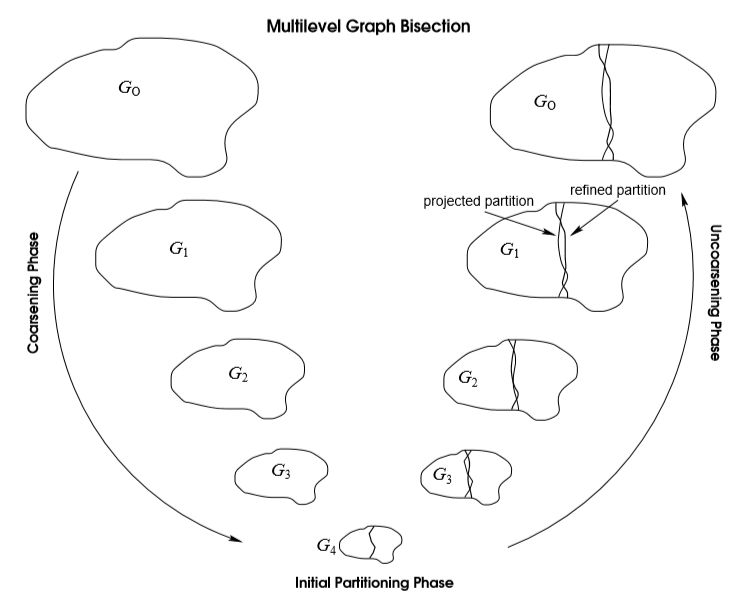
**Metis划分算法**

Metis是一种多级k路划分算法。图G=（V，E）首先被粗化（Coarsening）成小规模的图，其中包含数量较少的点。然后对粗化后的图进行k路划分，这时由于进行过粗化，点的数量大大减少，因此现在需要划分的是一个小得多的图，时间复杂度大大降低。在划分结束后，各个子图将被逐级细化（refinement），直到恢复最初点的数量，得到原始的图。粗化，划分和细化这三个步骤如图所示：



**粗化（Coarsening）**

在粗化的步骤里，通过点的融合，可以从原始图G0=（V0，E0）得到一组更小的图G1=（V1，E1）。在多数的粗化过程中，图Gi的一组点集会组合成下一级粗化图Gi+1中的一个点。令Viv为图中组合成图Gi+1中点Vi的点集。为了能使划分粗化后的图时能更好地考虑到原始图的点和权值，点v的权值会被设置成Viv中所有点权值的和。同时，为了在粗化图中保持原图的所有连接信息，与顶点v有关的边是Viv中边的集合。如果Viv中的多个点同时有边指向顶点u，那么顶点v到u的边权值为这些边权值的总和。这种粗化方法保证了粗化后的图具有如下特性：（1）划分粗化图时的边割与原图做同样划分时的边割相等；（2）对于粗化图的平衡划分可以使得原图的划分同样平衡。

上述的边融合方法的正式定义是匹配。一个图的匹配是一组边的集合，同一个集合中的边都指向不同的顶点。这样，由Gi生成的下一级粗化图Gi+1，可以通过融合集合中每条边的两个顶点来完成。没有匹配到的顶点则直接复制到Gi+1。通过这种顶点融合的方法，原始图中的顶点数量会大大减少。由于顶点融合的最终目的是为了减小原始图的规模，因此顶点的匹配应该最大化。最大匹配的定义如下：如果一个匹配在不使两条边指向同一个顶点的情况下无法再增加新的边，这个匹配叫做图的最大匹配。（由于匹配的计算方法不同，最大匹配的结果可能会有差异。）

下面将介绍两种选择最大匹配的方法，RM随机匹配和HEM权重边匹配。

RM：用随机算法可以快速有效地生成一个匹配随即最大匹配算法按照如下步骤工作：先按照随机的顺序遍历原始图中所有的顶点，如果一个顶点u还没有被匹配，算法会随机地选择其尚未匹配的相邻顶点。如果存在这样的相邻顶点v，就将边（u，v）加入匹配中。如果己经不存在未配的相邻顶点，则顶点u标记为未匹配顶点。

HEM：随机匹配是一种简单有效的求取最大匹配的方法，它使用了类似贪心算法的方法，在求得最大匹配的同时使粗化的级数最小化。然而，进行粗化的最终目的是为了在划分时得到最小的边割。考虑Gi=（Vi，Ei），用来粗化Gi的匹配Mi和它的粗化图Gi+1=（Vi+1，Ei+1）。如果A是边的集合，定义W（A）为A中边的权值的总和。显然可以得到下式：

W(Ei+1)=W(Ei)-W(Mi)

这样，与粗化之前相比，图减少的边的权值就等于匹配中所有边的权值总和。因此，通过选择边权值较大的匹配，就可以在粗化图中消减更大的权值。由于粗化图有更小的边权值，也就会有更小的边割。

寻找权重边匹配使用的随机算法与前面介绍的计算随机匹配的算法相类似。与随机匹配一样，所有顶点也是通过随机算法进行遍历。但是在这里并不是直接随机匹配顶点的任意一个相邻顶点，而是在所有相邻未匹配顶点v中寻找顶点使得边(u,v)权值最大。需要注意的是，这种匹配方法无法保证最终得到的匹配（在所有可能的的匹配中）有最大的边权值。

**初始划分（Initial Partitioning）**

Metis第二步是对粗化后的图进行k路划分。对粗化图Gm=（Vm，Em）计算划分Pm使得划分后的每部分大致均勾地含有原图的|V|/k个顶点。

一种产生k路初始划分的办法就是不断地对原图进行粗化操作，直到粗化图只剩下k个顶点。这个含有k个顶点的粗化图可以作为原始图的k路初始划分。在这个过程中会产生两个问题：1. 对于很多图来说，在进行了几次粗化之后，每一次的粗化过程所能减少的图的规模过小，因此粗化耗费的资源会很大。2. 即使将原图粗化到了仅剩个顶点，这些顶点的权值也极有可能差异很大，最终导致初始划分的平衡度大大降低。

多级划分算法是算法的基本思想，在进行了顶点融合算法之后，原图的规模已经减小到了比较容易处理的地步。经过顶点粗化的处理以后，顶点数和边数大大减少的新图将更有利于初始划分的进行。在多级划分中，初始划分就是现将图进行二路划分，并在划分的同时考虑负载的平衡。初始划分常用的算法是几何划分和谱系划分， 在我们这里采用KL算法进行初始划分。

**细化（Refinement）**

在这个步骤中，粗化图的划分，会通过回溯每一级的粗化图Gm的划分Pm还原成原图。由于Gm的每个顶点v都含有图Gi顶点集合Viv的一个独立子集，因此从Pi还原到Pi+1只需要将Viv中对应的点分配到Pi+1[v]中。例如Pi[u] =Pi+1[v]

虽然Pi+1是Gi+1的局部最小划分，但是细化后的划分Pi可能不再是Gi的局部最小划分。由于Gi更加精细，所以会有更大的自由度优化Pi以减少边割。因此，仍然可以使用局部细化启发式算法来优化划分Gi+1的划分。每进行一次细化，算法会对细化后的划分使用优化算法。划分优化算法的最基本思想是在划分后的两个部分中选择两个顶点集合进行互换，如果得到的新划分有更小的边割，则采用新的划分。