

类似编程语言的法术编排是 Noita 中的一大亮点，也是一大难点。[《如何在<Noita>中打造爆款法术》](#)（或曰《<Noita>法术编程指南》）一文较完整地讨论了法术编排和释放的机制，但依然存在不少失误。本文尝试在此基础上进行进一步的假设和实验，力图提出一个更加普适、更加易于理解和分析的理论，为 Noita 的游玩提供帮助。

讨论范围

本文将着重讨论以下几类法术：

- **投射物** (projectile)：既包括从法杖中飞出后拥有初速度的法术，即火花弹、炸弹、链锯之类一般意义上的投射物；也包括一般情况下不拥有初速度的法术，即粘液雾、静止之环、雷霆爆炸等静态投射物 (static projectile)。



- **投射修正** (projectile modifiers)：单独放置于法杖中不提供任何效果，需要和投射物以一定的先后顺序搭配，才能依附投射物产生效果的法术。如伤害增强、火焰轨迹、物质吞噬者、反重力等。



- **多重施放** (multicast)：增加施放数量的法术。可以让一根法杖一次射出更多的投射物，或者人一个带有触发/定时的投射物施放更多的投射物。如x重法术，x重散射法术和阵型。



以下法术不会深入讨论：

- **被动** (passive)：如能量盾、火把等，安装在法杖内无须施法即可在法杖处生效的法术。即使如此，它们依然会被以类似投射修正的方式在法杖内被触发。因此，可将它们视为对法杖起作用的投射修正看待。
- **实用** (utility)：如全知之眼，类似静态投射物，但仅仅在施法的一瞬间提供实用功能。
- **材料** (material)：如点金 (Touch of Gold)、酸液之海、水等，类似静态投射物，但仅仅在施法的一瞬间在世界内直接生成/转化物质。



关于魔杖

- 法杖的施放延迟、充能时间不会讨论。
- 始终施放的效果是：每次施法时最先施放始终施放的法术，并且不消耗魔力 and 施放数

另外，本文不会过多解释各种法术的效果和使用时的注意事项、搭配技巧。

分析方法的改进

要分析法术的不同编排所产生的效果，首先需要将法杖内的法术分割成有意义、相互关联的部分，从而化整为零，方便理解。

《<Noita>法术编程指南》提出了将法术拆分成层次明显的树状图，并根据一定原则加以分析的思路。

投射修正法术三原则

- 投射修正法术不修正投射物(包括静态投射物) 以外的其他法术,例如投射修正法术自身、物质法术、被动法术等。
- 如果投射修正法术是某法术的父法术,那么二者之间有其他触发法术时不产生修正;如果投射修正法术是某法术的子法术,那将不产生修正。
- 投射修正法术上述情况之外的任何法术。

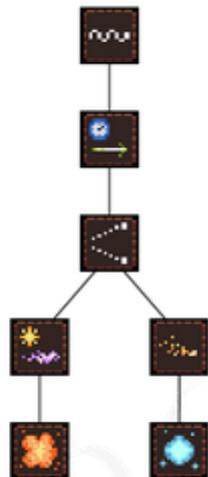


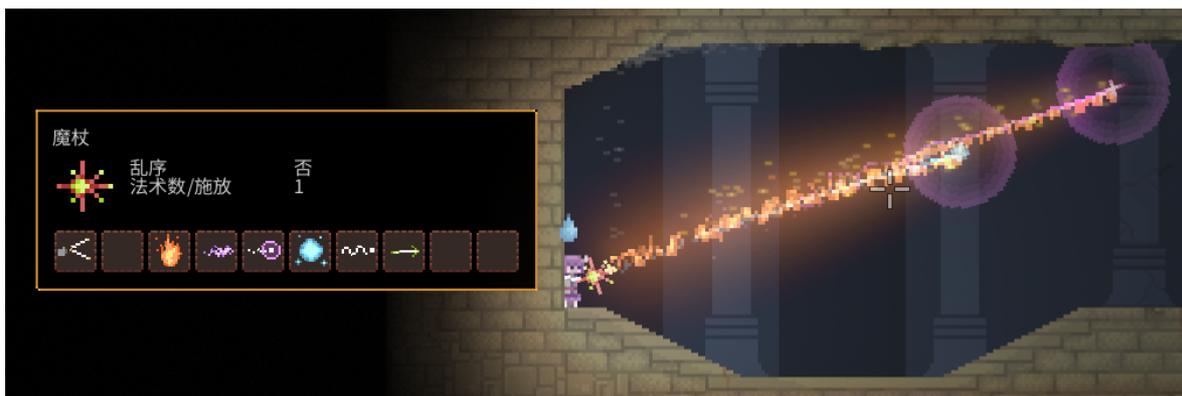
图 7: 对应的法术树

图中的火焰轨迹应当同时修正能量球和带有触发的火花弹，然而在图中它与火花弹处于两条不同的分支，并且处于能量球的上方，容易让人误以为能量球是火焰轨迹的子法术，火花弹与火焰轨迹不相干，不接受修正。另外，这个法术树可以被搭建多种完全等效的结构：火焰轨迹在带有触发的火花弹处，而不在能量球一侧；或者干脆将火焰轨迹置于阵型-分叉的上方。

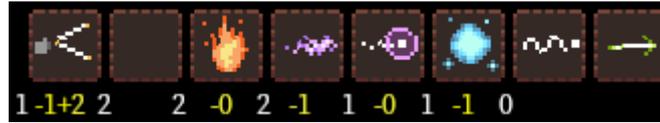
为了减少误导性，简化分析过程，这里提出了一种围绕触发/定时法术展开的，贴合横向编排思路的分段分析方式。

施法数的变化

观察如下序列在编辑后的**第一次发射效果**，可以看到只射出了有“燃烧轨迹”和“伤害立场”的火花弹和能量球体，没有“蛇形”的效果，没有发射魔法箭。



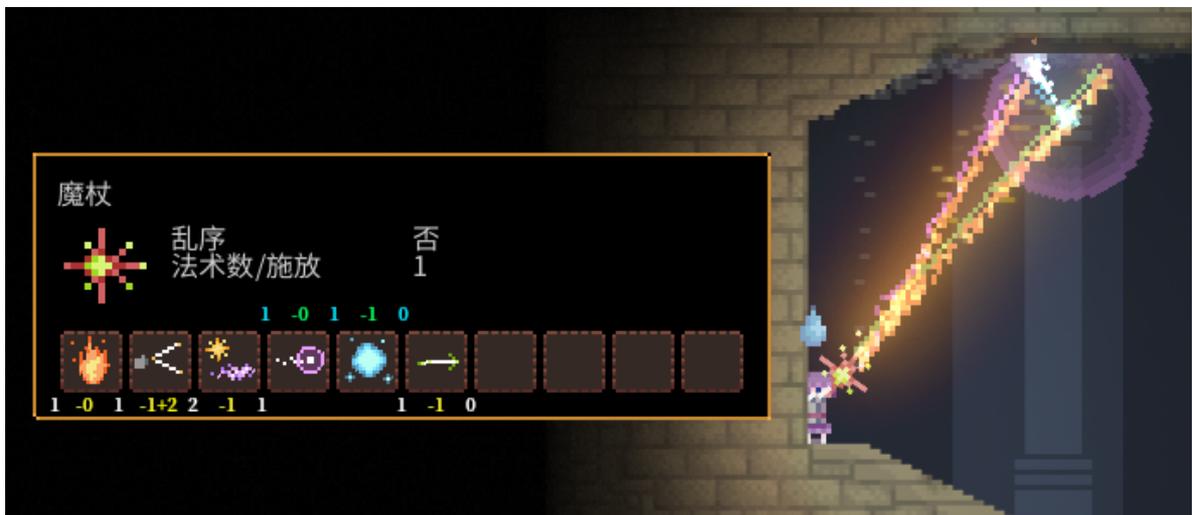
如果按照从左到右的思维去理解，可以认为法杖在“二重散射法术”这里消耗了法杖自带的 1 施放数，并且增加了多重施放的 2 施放数。随后从左到右涉及到燃烧轨迹、火花弹、伤害力场、能量球体。到能量球体处，施放数量被两个投射物法术消耗殆尽，于是后续的蛇形和魔法箭没有涉及到此次施法。



以上是施法数的变化规律。可以看出，**投射修正不消耗施法数，其它法术都会消耗 1 施法数，多重施放会在消耗 1 施法数的基础上再增加若干施放数。**

触发与定时法术

带有触发/定时的投射物法术时，触发/定时法术会携带一部分法术。这些**触发/定时法术会独立计算施放数**。

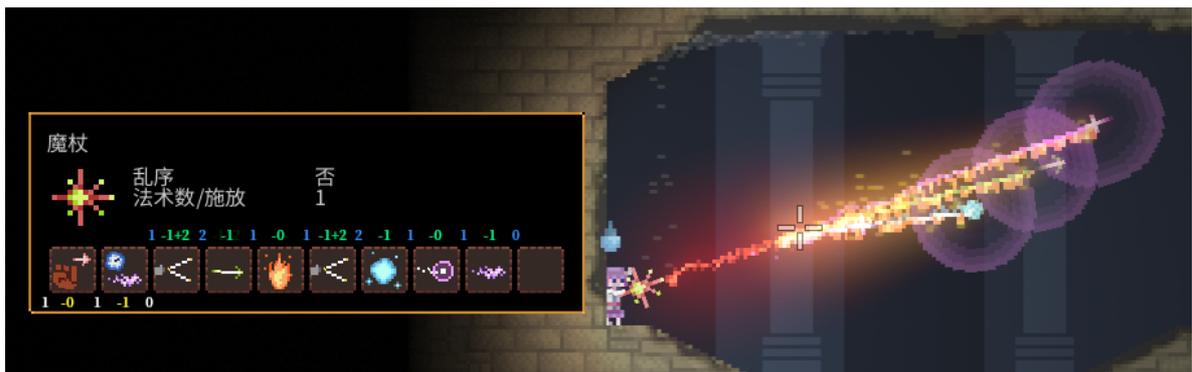


在上面的例子中，“带有触发的火花弹”单独对施放数进行计算，从而能够在远处施放伤害力场和能量球体。这个行为使得**外部的施放数在进行计算时跳过了伤害力场和能量球体**，直接对魔法箭进行计算。

另外，燃烧轨迹只对火花弹和魔法箭有效；伤害力场只对能量球体有效。这时候我们需要对投射修正的作用范围进行考虑。

投射修正的范围

当同时考虑多个投射修正、多重施放和带有触发/定时的法术时，情况会稍显复杂。为了探明这一点，我们需要对下面这个序列进行分析：



如果直接进行观察，很可能会误以为燃烧轨迹只对能量球体和火花弹有效。然而在分析后会发现，魔法箭、能量球体和火花弹都会享受燃烧轨迹和伤害力场的投射修正效果。可以说，**多重施放只会造成施放数的增减，并不会产生如同定时/触发法术那样的嵌套关系。**

并且，由于沉重一击只对带有定时的火花弹有效；燃烧轨迹和伤害力场只对魔法箭、能量球体和火花弹有效。再结合之前的几个例子，可以推断出**投射修正对且仅对嵌套关系中同一层的所有法术生效**。在**触发与定时法术**段落中的例子里，由于魔法箭和带有触发的火花弹同属于施放数计算的最外层，都会享受到处于同一层的燃烧轨迹。



快速分析法

根据之前提到的几个要点，这里归纳总结出一个快速分析法术嵌套关系、投射修正作用范围的方法：

1. 投射修正不消耗施法数，其它法术都消耗 1 施法数，多重施放会在消耗 1 施法数的基础上再增加若干施放数。
2. 定时/触发法术会造成嵌套关系。不同嵌套层的施放数分别计算。
3. 多重施放增加的施放数作用与它所处的嵌套层。
4. 内层的施放数计算完毕后，外层的施放数应该跳过内层部分计算。
5. 投射修正对且仅对所处的嵌套层内的所有投射物起效。

根据快速分析法，在《<Noita>法术编程指南》中提到的法术序列，可以在快速计算完施放数后，拆分成如下三部分：



对分段进行整理，法术序列最终变成了这样的嵌套关系，一目了然：



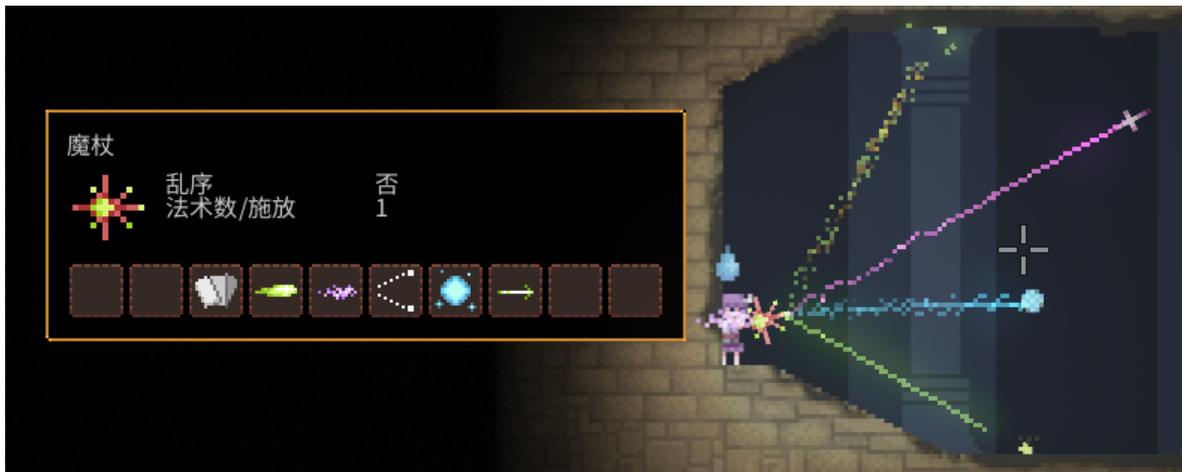
大家都是投射修正!

多重施放可以当投射修正



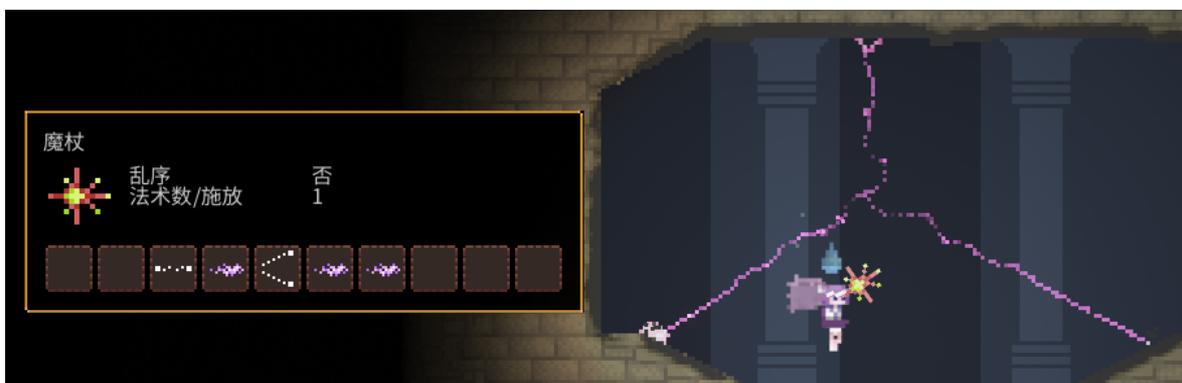
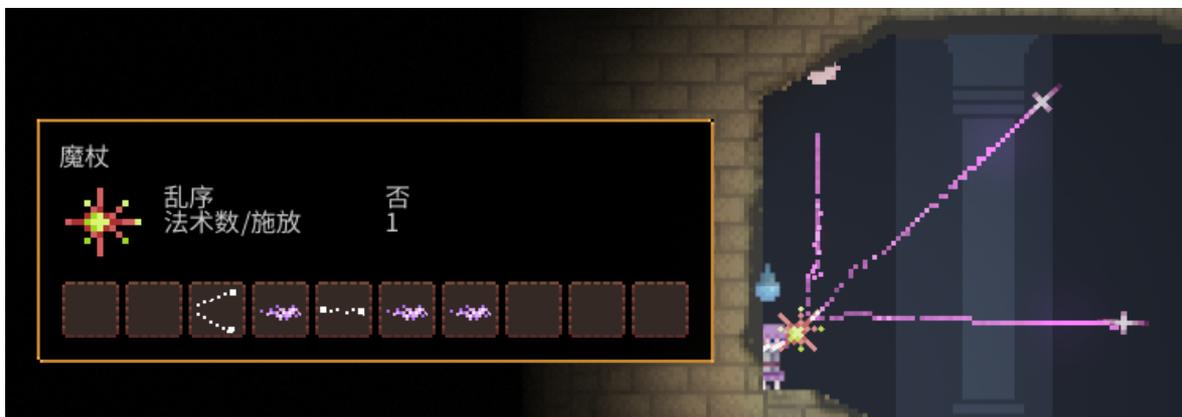
既然多重施放并不造成嵌套关系，那么阵型法术并不能将后方的法术单独囊括起来，组成一定的形状。

因此，阵型类法术实际上以类似投射修正的方式在一定角度内均匀顺时针分布所处嵌套层中所有的投射物。例如，阵型-分叉能够作为一个投射修正，被修正的投射物在 $\pm 45^\circ$ 的范围内顺时针均匀施放：



以此类推，阵型-前后、阵型-五边形、阵型-六边形是在 $\pm 180^\circ$ ，即一整圈的范围均匀分布投射物，阵型-上下的范围则是 $\pm 180^\circ$ 。阵型-三叉的范围比阵型-分叉的范围稍小一点，约为 $\pm 30^\circ$ 。

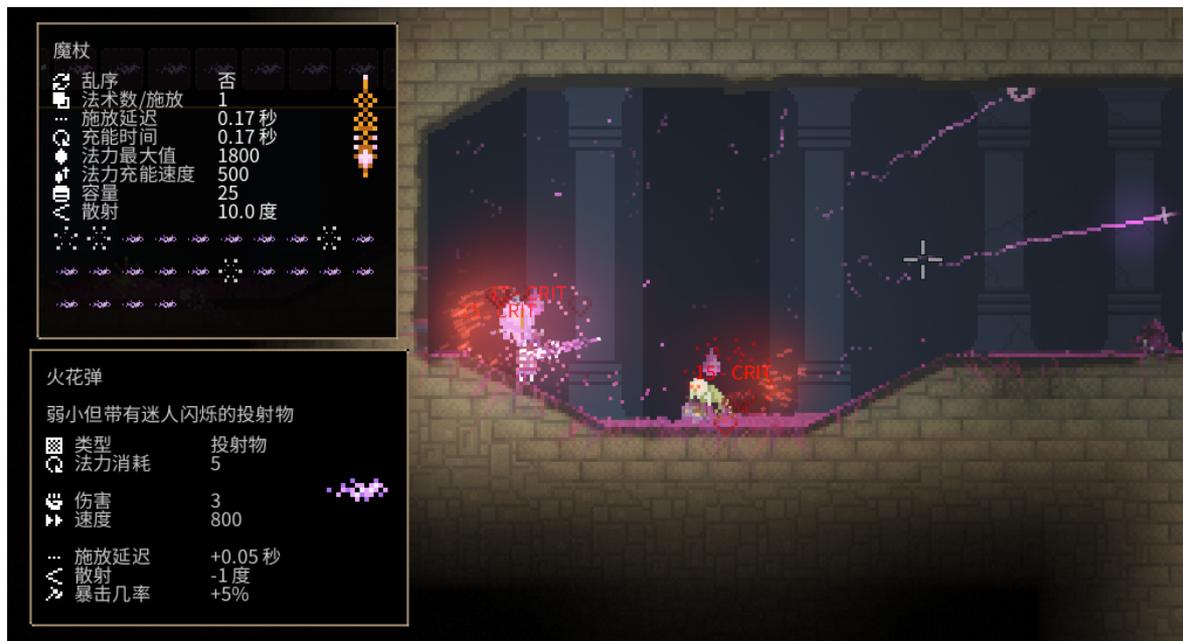
当有多个阵型法术同时存在时，最左边的法术决定分布范围：



投射物也可以当投射修正

事实证明，投射修正的分类并没有那么明确——部分投射物也能提供类似的效果。

例如玩家们喜闻乐见的火花弹，就提供了 +5% 暴击和 -1 度散射的修正效果。而这些修正效果，是对同一嵌套层内所有法术起效的。因此，当我们用丧心病狂的 20 个火花弹填满一个法杖，并让它们一次性发射出来的时候，每一发火花弹都有着 $20 * 5\% = 100\%$ 的暴击加成，并且总共有 $20 * (-1^\circ) = -20^\circ$ 的散射修正。每一发都精准而致命：



法术池的透支

《<Noita>法术编程指南》一文中也提到了“法术池”这个概念：在切换法杖和编辑法术时，法杖会刷新它的“法术池”。有序法杖的法术池中全部法术按照玩家设置的顺序排列；而乱序法杖的法术池中的法术是随机打乱的。游戏施法时并不直接从法杖中读取法术，而是从法术池读取。

可以发现，如果一次施法用尽了法术池，则魔杖会立即生成新的法术池继续施法，同时魔杖开始充能。但是如果一次施法第二次用尽了法术池，施法将强制终止.....

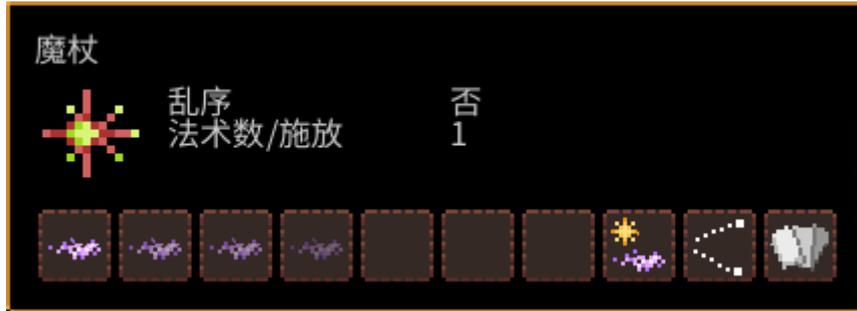
——《<Noita>法术编程指南》

在法杖中放入两只带有定时的光剑。当游戏处理到第二个光剑时会发现：光剑还可以继续嵌套，而法术池已经耗尽。如果编程指南中的这段话属实，那么游戏会生成新的法术池，再次以相同的顺序供应两个光剑，用尽第二个法术池，强制终止施法。最终效果会是长度为四倍的光剑。然而，实际情况下射出的却是长度为二倍的光剑：



有限的透支

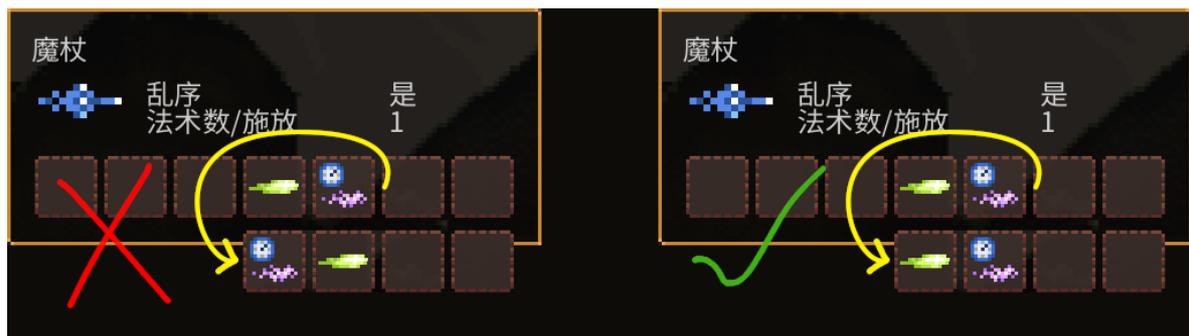
通过在以下法杖最前端填充 1~4 个火花弹并进行实验，会发现：带有触发的火花弹进行透支法术池时，总是会恰好发射出前端的所有火花弹；并不会透支到带有触发的火花弹本身。由此可以得出，**每次施放的总法术数量不得超过法术池的法术数量。**



法术池何时刷新？

如果在一只乱序法杖中放入魔法弹和带有定时的火花弹，按照编程指南的理论，如果第一个法术池是 {魔法弹，带有定时的火花弹}，在射出单发魔法弹后，下一次射出带有定时的火花弹时会进行透支。如果在透支前刷新了法术池，由于法杖时乱序的，将有概率刷新出 {带有定时的火花弹，魔法弹} 这样的法术池。结果是会出现两个嵌套的带有定时的火花弹。然而事实上并不会出现这样的射击效果。

因此这里可以推导出关于法术池的第二条规律：**当法术池透支出现时，游戏并不会立刻刷新法术池，而是从头再次使用原来的法术池。在计算完施放的所有法术后，再刷新法术池。**



用投射修正射出子弹！

以下是一个简单却有实用性的例子：

第一次点击鼠标，法杖会施放出带有加速修正的霰弹，法术池剩余一个降低散射。

第二次点击鼠标，由于降低散射是投射修正，不减少施放数，游戏透支法杖，从头获取加速修正和霰弹。第二次的法术组合是 {降低散射，加速，霰弹}。之后，法杖重新充能。

这样编排法杖的好处量两个：如果充能时间大于施放延迟，适当将投射修正置后可以利用法术池透支机制，实现每次充能可以设计两次的效果。另外，这样编排可以让霰弹第一发保持散布，第二发降低散射，集中火力，并且同时具备加速的修正效果，可以同时打击体积小、数量多和高生命、数量少的怪物。



综合性例子

这里举一个综合性较强的例子，完整阐述一下这篇文章提到的机制：



首先注意到法杖的施放数是 2，因此第一次施法会毫无悬念地发射出一发霰弹和一发火花弹（没有加速修正）。

第二次施法从加速修正开始。三重施放与阵型-背后使定时火花弹的施放数量达到 $1 + (-1 + 3) + (-1 + 2) = 4$ ，但是定时火花弹的右侧只有三个普通火花弹，于是需要从头透支法术池。目前施放了 8 个法术，法术池有 10 个法术，因此可透支上法术池最开始的一颗火花弹，于是定时火花弹攒齐了四个投射物法术。

这时候不要忘了法杖的施法数是 2。由于目前施放了 9 个法术，可以继续“透支”法术池。再加上霰弹，法杖的两个施法数也被满足了。

将第二次施放的法律序列按照嵌套关系分段后，效果如下图所示：



最后统计一下所有的投射修正：法杖直接发射的定时火花弹与霰弹都是有加速修正和火花弹的 +5% 暴击修正的；定时火花弹触发的四个火花弹都有沉重一击修正和四个火花弹提供的 $4 * 5\% = 20\%$ 的暴击修正。另外，阵型-背后使得这四个火花弹会在 $\pm 180^\circ$ 内均匀散布。

施放效果：

