arXiv:2603.04376v1 [math.AC] 4 Mar 2026

Lean 中忠实平坦下降射影性

LIRAN SHAUL

摘要。我们在 Lean 中形式化了交换代数中的以下奠基性结果：设 $R \to S$ 是交换环之间的忠实平坦映射（这些环不一定是诺特环），且 P 是任意的 R-模。那么，P 在 R 上是投射模，当且仅当 $S \otimes_R P$ 在 S上是投射模。这一形式化工作验证了 Perry 对 Raynaud 和 Gruson 经典研究中一个微妙漏洞的修正；该结果是研究交换诺特环有限维数的关键要素。

0. 引言

奠基性论文 [7] 是 20 世纪交换代数领域最具影响力的论文之一。它的一项重大成就是解决了 Bass 在[2], 提出的问题：任何交换诺特环的（大）投射有限维数都等于其克鲁尔维数。用不那么专业的语言来说，这意味着，对于交换诺特环 A，任何可以用有限个（规模可能无限的）矩阵表示的模，都可以用至多 $\dim(A)$个矩阵表示。

为了证明这一结果，和现代交换代数中许多问题的惯常做法一样，作者分三步进行。首先，在完备诺特局部环的情形下证明该结果；然后，利用忠实平坦性将其推广到任意诺特局部环；最后，通过局部到整体论证，证明该结果对所有交换诺特环都成立。对于这个问题来说，这三步都极具挑战性。

从完备局部环推广到任意诺特局部环的步骤包含在以下定理中：

定理 I. 设 $R \to S$ 是交换环之间的忠实平坦映射，且 P 是一个R模。那么，P 是 R 上的投射模，当且仅当 $S \otimes_R P$ 这个 S模是 S上的投射模。

注意，这里并未假设这些环是诺特环，而且 P 可以是无限生成的。

正如 Gruson 后来指出的，[3], 定理I 在 [7] 中的证明存在一个漏洞。
定理的完整证明 I 见于 [6, 定理 9。6], 并且也已纳入 Stacks 项目中的 [1, 标签 05A9], 但据我们所知，它从未经过同行评审。
本文旨在介绍这一结果近期在 Lean 中成功形式化的过程，以及其所有前置内容。所有 Lean 代码均位于 [8]。

数学主题分类 2020：13C10、16D40、68V20。

定理 I 的形式化需要系统地发展模范畴中的推出、借助支配判据得到的普遍单射性、对逆系统的 Mittag–Leffler 条件进行完整的形式化处理，以及使用序数的超限分层框架。Lean 4 此前并未提供这些组成部分，因此我们从零开始开发，最终编写了超过 10,000 行代码。

Mittag-Leffler 模的概念最初由[7]定义并加以研究。我们在本项目中对这一概念、其基本性质及各种应用进行了详尽的形式化。或许我们形式化的最有趣结果（定理 I 之外）是以下结果，它恰好是 [6, 定理 7。4]:

定理 II。 设 R 是一个环， P 是一个左 R-模。那么， P 是 R 上的投射模，当且仅当满足以下条件：

(1) P 是 R 上的平坦模。*(2)* P 是 R 上的 *Mittag-Leffler* 模。*(3)* P 是可数生成 R-模的直和。

这里，环 R 可以是非交换的。投射模满足条件 (3) 是 Kaplansky 的一个经典结果，我们也对其进行了形式化。证明还依赖于 Lazard 定理，该定理指出任意平坦模都是有限生成自由模的直极限；这一结果也已形式化。

在本文余下部分，我们将介绍形式化的最重要结果，包括它们在 Lean 中的精确陈述，以及形式化过程中遇到的一些挑战。这里介绍的所有结果均已使用 Lean 4 和 Mathlib 成功编译，我们计划将其提交以纳入 Mathlib。

致谢。
作者感谢 Mirek Olsak 以及 Zulip 社区提供的宝贵讨论。作者也感谢大型语言模型 Claude Opus 4.5 和 Claude Opus 4.6 协助生成本项目中使用的部分 Lean 代码。本工作获得了捷克科学基金会的 GA ČR 26-22734S 项目资助。

1. Kaplansky 逐层分解

本节的结果位于 [8] 中的文件 lib/Kap.lean。定理 I 和定理 II 的证明有一个关键环节，即将问题归约到可数生成模的情形。为此，我们需要以下由 Kaplansky 提出的结果 [4, 定理1], ，我们已将其形式化：

定理1.1。 设 R 是一个环，设 M 是一个 R-模，并假设 M 是可数生成 R-模的直和。若 P 是 R-模且为 M 的直和项，则 P 也是可数生成 R-模的直和。

为了在 Lean 中形式化这一结果，我们首先将由某个可数子集生成的模定义为可数生成：

```
def Submodule.IsCountablyGenerated
{R : Type u} {M : Type v}
[Ring R] [AddCommGroup M] [Module R M]
(N : Submodule R M) : Prop :=∃ s : Set M,
 s.Countable ∧ span R s = ⊤
```

为避免不必要地复杂使用商模（这里并非严格需要，因为我们只处理直和项），我们引入直和补的以下辅助定义

```
def IsRelativeComplement
  {R : Type u} {M : Type v}
  [Ring R] [AddCommGroup M] [Module R M]
  (A B C : Submodule R M) : Prop :=
  A ≤ B ∧ C ≤ B ∧ Disjoint A C ∧ A ⊔ C = B
```

使用这些定义后，我们仿效 Kaplansky，引入 Kaplansky 逐层分解的概念，其定义如下：

```
structure KaplanskyDevissage
     (R : Type u) [Ring R]
  (M : Type v) [AddCommGroup M] [Module R M]
  (S : Ordinal.{w}) whereseq : Set.Iio S → Submodule R M
  monotone : ∀αβ   : Set.Iio S, α≤β→    seq α≤  seq β
  zero_eq_bot (h : 0 < S) : seq ⟨0, ⟩h = ⊥
  union_eq_top : ⨆( α : Set.Iio S, seq α) = ⊤
  limit_continuity : ∀α  : Set.Iio S,
  Order.IsSuccLimit α.val →
  seq α = ⨆β  : {x : Set.Iio S // x.val < α.val}, seq β.val
  succ_step : ∀α ( : Ordinal.{w}) (h : Order.succ α < S),∃
  C : Submodule R M,IsRelativeComplement
  (seq ⟨α, (Order.lt_succ α).trans ⟩h)(seq ⟨Order.succ α, ⟩h)
  C ∧C.IsCountablyGenerated
```

因此，与 R-模 M 相关联的这类结构由某个序数 S以及一个大小为S的 M 的滤过给出，并且该滤过中的每个元素都是 M的直和项，且其中每个相继商模都是可数生成的。使用

```
seq : Set.Iio S → Submodule R M
```

而不是可能更常见的

```
seq : ∀α  : Ordinal, α < S → Submodule R M
```

目的是确保我们保持在相同的宇宙层级，否则 Lean 会忽略 $\alpha < S$条件，不必要地从宇宙 w 跳到 $w+1$。
接下来，我们沿着 [1, 标记 058V] 和 [1, 标记 058W] 证明，对于某个环 R上的 M-模， R 存在一个关于 M、相对于某个序数 S的 Kaplansky 逐层分解，等价于 M 是可数生成模的直和：

```
theorem kaplansky_devissage_iff_direct_sum
    {R : Type u} [Ring R] {M : Type v} [AddCommGroup M] [Module R M] :∃
    ( ι( : Type w) (fam : ι→  Submodule R M),
    DirectSum.IsInternal fam ∧∀  i, (fam i).IsCountablyGenerated)↔∃


     (S : Ordinal.{w}),  Nonempty (KaplanskyDevissage R M S)
```

该证明通过在两个方向上构造显式滤过来形式化这一等价性，其中较为困难的逆向需要采用两层归纳，首先将独立性归约为指标的有限集，通过

```
Submodule.mem_span_finite_of_mem_span
```

然后，利用滤过的不交条件，逐一剥离极大序数指标，从而消去各个贡献项。

利用这一刻画，我们接着证明定理 1.1。给定一个 R-模 $M = \oplus_{\alpha\in I} C_\alpha$，它是可数生成模 C_α的直和，以及 N的一个直和项 M。设 $N \oplus K = M$。目标是证明 N是可数生成 R-模的直和。利用

```
theorem kaplansky_devissage_iff_direct_sum
```

上文所述，只需为 N 构造一个 Kaplansky 逐层分解。仿照 [4,定理 1] 和 [1, 标签 058X], 中的论述，我们为 M 构造一个 Kaplansky 逐层分解，并要求它额外满足：对每个 α，都有

$$M_\alpha = (M_\alpha \cap N) \oplus (M_\alpha \cap K) \tag{1.1}$$

为此，选取指标集的一个良序 I。假设已定义了一个 M_α，我们首先取 $M_{\alpha+1}$ 为最小的 $j \in I$，使得 j。随后定义 $C_j \nsubseteq M_\alpha M_{\alpha+1}$ 为由 M_α、C_j的生成元，以及从 I 中其他指标对应的生成元所生成的模，从而使形如方程（1.1）的分解对 $M_{\alpha+1}$仍然成立。这只需要可数多个指标，因此 $M_{\alpha+1}/M_\alpha$ 是可数生成的。按照这种方法，我们得到如下形式的定理1.1：

```
theorem dirsummand_of_dirSumCountable {R : Type u} [Ring R]
{M N : Type v} [AddCommGroup M] [AddCommGroup N] [Module R M]
[Module R N] (exDecom : ∃ι ( : Type w) (fam : ι→  Submodule R M),
DirectSum.IsInternal fam ∧∀  i, (fam i).IsCountablyGenerated)
(i : N →ₗ[R] M) π( : M →ₗ[R] N) (πhi : π.comp i = LinearMap.id) :∃κ
( : Type w) (fam' : κ→  Submodule R N),DirectSum.IsInternal fam' ∧∀  j,
 (fam' j).IsCountablyGenerated
```

由于投射模是自由模的直和项，而自由模是循环模的直和，因此这一结果立即表明，任意投射模都是可数生成投射模的直和。由此我们得到:

```
theorem proj_is_dirSumCountable {R : Type u} {P : Type u} [Ring R]
[AddCommGroup P] [Module R P] (pproj : Module.Projective R P)
: IsDirectSumOfCountablyGenerated R P
```

2. 拉扎德定理

本节的结果位于 [8]中的文件 lib/Lazard.lean。根据上一节的结果，定理 II 的证明随即归结为证明一个可数生成的平坦模若满足 Mittag-Leffler 条件，则它是投射模。这需要用到经典的拉扎德定理 [5]:

定理 2.1。设 R 是一个环，F 是一个 R-模。那么，F 是平坦的，当且仅当 F 同构于有限表示自由 R-模的一个直极限。

与此密切相关，而且对于处理 Mittag-Leffler 条件也至关重要的，还有一个更基本的事实；这个事实在 Mathlib 中也尚未收录，即任意环上的任意模都是有限表示模的直极限。
我们先证明这个更基础的结果，表述如下:

```
theorem Module.isDirectLimit_of_finitelyPresented :∃ι
       ( : Type u) (_ : Preorder ι) (_ : IsDirected ι (· ≤ ·))
    (_ : Nonempty ι) (_ : DecidableEq ι) (G : ι→  Type u)
    (_ : ∀ i, AddCommGroup (G i)) (_ : ∀ i, Module R (G i))
    (_ : ∀ i, Module.FinitePresentation R (G i))
    (f : {i j : ι}→  i ≤ j → G i →ₗ[R] G j)
    (_ : DirectedSystem G (fun _ _ h => f h)),
    Nonempty (DirectLimit G (fun _ _ h => f h) ≃ₗ[R] M)
```

接下来我们将证明 Lazard 定理。为此所需的一个关键技术引理如下：

引理 2.1. 设 R 是一个环，M 是一个平坦左 R-模。设 $J \subset M \times \mathbb{Z}$ 是一个有限集，$N \subseteq R^{(J)}$ 是一个有限生成的 R-子模。设 $\psi_J : R^{(J)} \to M$ 是由 $e_{(m,z)} \mapsto m$ 定义的 R-线性映射。
若 $N \subseteq$核(ψ_J)，则存在一个包含 J 的有限集 J'，以及一个有限生成子模 $N' \subseteq R^{(J')}$，使得：

> *(1)* $N' \subseteq$ 核$(\psi_{J'})$。*(2)* 商模 $R^{(J')}/N'$ 是一个自由 R-模。*(3)* 在自然嵌入 $\iota : R^{(J)} \hookrightarrow R^{(J')}$ 下，我们有 $\iota(N) \subseteq N'$。

我们证明如下：

```
lemma Module.Flat.enlarge_to_free [Module.Flat R M]
{J : Finset (M × ℤ)} {N : Submodule R (J →₀ R)} (hN : N.FG)
(hN_ker : ∀ x ∈ N, Finsupp.embDomain ⟨Subtype.val,
Subtype.⟩val_injective x ∈
LinearMap.ker (Finsupp.linearCombination R (Prod.fst : M × ℤ→  M))) :∃
(J' : Finset (M × ℤ)) (hJJ' : J ⊆ J') (N' : Submodule R (J' →₀ R))
(_ : N'.FG), ∀( x ∈ N', Finsupp.embDomain ⟨Subtype.val,
Subtype.⟩val_injective x ∈
LinearMap.ker (Finsupp.linearCombination R (Prod.fst : M × ℤ→  M)))∧
Module.Free R ((J' →₀ R) / N') ∧∀ x ∈ N,
 Finsupp.lmapDomain R R (fun i => ⟨i.1, hJJ' i⟩.2) x ∈ N'
```

这使我们能够构造出具有所需性质的有向系统，并得到定理2.1 的如下形式：

```
theorem Module.Flat.Lazard :
    Module.Flat R M ↔∃ ι
       ( : Type u) (_ : Preorder ι) (_ : IsDirected ι (· ≤ ·))
    (_ : Nonempty ι) (_ : DecidableEq ι) (G : ι→  Type u)
    (_ : ∀ i, AddCommGroup (G i)) (_ : ∀ i, Module R (G i))
    (_ : ∀ i, Module.FinitePresentation R (G i))
    (_ : ∀ i, Module.Free R (G i)) (f : {i j : ι}→  i ≤ j → G i →ₗ[R] G j)
    (_ : DirectedSystem G (fun _ _ h => f h)),
    Nonempty (DirectLimit G (fun _ _ h => f h) ≃ₗ[R] M)
```

3. 模的推出

本节的结果位于文件 lib/Pushout.lean 中， [8]。从现在起，我们专门讨论环 R 为交换环的情形。若 R 是一个环，且 $f : A \to B$ 和 $g : A \to C$是两个 R-线性映射，定义在 R-模之间，则它们的推出是一个 R-模 D 以及一对映射 $f' : B \to D$ 和 $g' : C \to D$，使得图表

$$\begin{array}{ccc} A & \xrightarrow{f} & B \\ {\scriptstyle g}\downarrow & & \downarrow{\scriptstyle f'} \\ C & \xrightarrow[g']{} & D \end{array}$$

它是交换的，并且对于这一性质具有普遍性。在 Lean 中工作时，与其使用这样的普遍构造，不如选择推出的一个具体实现，这当然更简单。可以取
$D = (B \oplus C)/T$，其中 T 是由集合生成的 $B \oplus C$ 的子模

$$\{(f(a), -g(a)) \mid a \in A\}.$$

映射 f' 和 g' 则由包含映射 $B \hookrightarrow B \oplus C$ 和 $C \hookrightarrow B \oplus C$ 与自然满射 $B \oplus C \to (B \oplus C)/T$ 复合而得。
我们使用以下定义在 Lean 中实现这一点。子模 T 定义为

```
def pushoutSubmodule (f : A →ₗ[R] B) (g : A →ₗ[R] C) : Submodule R (B × C)
:=  Submodule.span R (Set.range (fun a => (f a, -g a)))
```

于是推出为

```
def Pushout (f : A →ₗ[R] B) (g : A →ₗ[R] C) : Type u :=
    (B × C) / pushoutSubmodule f g
```

并且映射 f' 和 g' 由下式给出

```
def Pushout.inl (f : A →ₗ[R] B) (g : A →ₗ[R] C) : B →ₗ[R] Pushout f g :=
    (Submodule.mkQ _).comp (LinearMap.inl R B C)

def Pushout.inr (f : A →ₗ[R] B) (g : A →ₗ[R] C) : C →ₗ[R] Pushout f g :=
    (Submodule.mkQ _).comp (LinearMap.inr R B C)
```

然后我们证明推出所满足的各种基本性质，包括万有性质、某种函子性，以及它与张量积交换这一事实。更精确地说，成立如下结论：

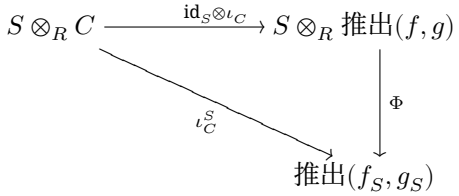
定理3.1. 设 R 为一个交换环， S 为一个 R-代数，且 $f : A \to B$、 $g : A \to C$ 为 R-线性映射。记

$$f_S = \mathrm{id}_S \otimes f : S \otimes_R A \to S \otimes_R B, \qquad g_S = \mathrm{id}_S \otimes g : S \otimes_R A \to S \otimes_R C$$

基变换映射。则存在一个 S-线性同构

$$\Phi : S \otimes_R \mathrm{Pushout}(f, g) \xrightarrow{\sim} \mathrm{Pushout}(f_S, g_S).$$

此外，下列图表交换：



其中， $\iota_C : C \to$ 推出(f, g) 和 $\iota_C^S : S \otimes_R C \to$ 推出(f_S, g_S) 是分别映入相应推出的典范右含入。

我们在 Lean 中使用以下内容来实现：

```
noncomputable def Pushout.baseChangeEquiv
    {R : Type u} [CommRing R]
    {S : Type u} [CommRing S] [Algebra R S]
    {A B C : Type u} [AddCommGroup A] [AddCommGroup B] [AddCommGroup C]
    [Module R A] [Module R B] [Module R C]
    (f : A →ₗ[R] B) (g : A →ₗ[R] C) :
    S ⊗[R] Pushout f g ≃ₗ[S]
    Pushout (LinearMap.baseChange S f) (LinearMap.baseChange S g)
```

and

```
lemma Pushout.baseChangeEquiv_inr
    {R : Type u} [CommRing R]
    {S : Type u} [CommRing S] [Algebra R S]
    {A B C : Type u} [AddCommGroup A] [AddCommGroup B] [AddCommGroup C]
    [Module R A] [Module R B] [Module R C]
    (f : A →ₗ[R] B) (g : A →ₗ[R] C) :↑
    ((Pushout.baseChangeEquiv (S := S) f g) ∘ₗ
    LinearMap.baseChange S (Pushout.inr f g)) =
      Pushout.inr (LinearMap.baseChange S f) (LinearMap.baseChange S g)
```

4. 普遍单射映射

本节的结果位于文件 lib/UnivInj.lean 中， [8]给定一个交换环R，回顾一下，一个 R模 $f : M \to N$ 的映射称为普遍单射映射，如果对于任意 R模 Q，映射 $f \otimes_R \mathrm{id} : M \otimes_R Q \to N \otimes_R Q$ 都是单射。这样的映射必然是单射。在 Lean 中，我们将其定义为:

```
def UniversallyInjective (f : M →ₗ[R] N) : Prop :=∀
(Q : Type u) [AddCommGroup Q] [Module R Q],
Function.Injective (f.rTensor Q)
```

普遍单射映射满足的一项关键性质，是我们需要的以下提升性质，该性质包含在 [1, tag 058K] 中:

定理 4.1。 设 R 为交换环，设 $f : M \to N$ 为普遍单射的 R-线性映射，且 F, G 为有限生成自由 R-模。假设存在 R-线性映射 $g : F \to M$、 $h : G \to N$ 和 $k : F \to G$，使得该方块

$$\begin{array}{ccc} F & \xrightarrow{k} & G \\ {\scriptstyle g}\downarrow & & \downarrow{\scriptstyle h} \\ M & \xrightarrow[f]{} & N \end{array}$$

交换，即 $h \circ k = f \circ g$。那么存在一个 R-线性映射 $\phi : G \to M$，使得

$$\phi \circ k = g.$$

我们将其形式化为:

```
theorem universially_injective_lift_free
{R : Type u} [CommRing R]
{M : Type u} {N : Type u}
[AddCommGroup M] [AddCommGroup N]
[Module R M] [Module R N]
{F : Type u} [AddCommGroup F] [Module R F]
[Module.Free R F] [Module.Finite R F]
{G : Type u} [AddCommGroup G] [Module R G]
[Module.Free R G] [Module.Finite R G]
(f : M →ₗ[R] N)
(univf : UniversallyInjective f)
(g : F →ₗ[R] M)(h : G →ₗ[R] N)(k : F →ₗ[R] G)
(commSquare : h ∘ₗ k = f ∘ₗ g) :∃φ: G →ₗ[R] M,
 φ∘ₗ  k = g
```

证明的关键思路是利用 F 和 G 的选定基选取坐标，将提升问题化为寻找满足以下条件的 $z_j \in M$：对于 F 的所有基向量 e_i，都有 $x_i = \sum_j a_{ij} z_j$；其中，$x_i = g(e_i)$ 和 a_{ij} 是由 $k(e_i) = \sum_j a_{ij} b_j$ 定义的 k 的矩阵系数。然后利用普遍单射性，得出这样的 z_j 存在。
另一个关键性质是，这一概念可沿忠实平坦映射下降：

定理 4.2。设 R 是一个交换环，$R \to S$ 是交换环之间的忠实平坦映射，$f : M \to N$ 是一个 R-线性映射。若其基变换映射

$$f_S = \mathrm{id}_S \otimes f : S \otimes_R M \to S \otimes_R N$$

作为 S-线性映射是普遍单射的，则 f 作为 R-线性映射是普遍单射的。

我们将其形式化如下：

```
theorem UniversallyInjective.of_baseChange_faithfullyFlat
    {R : Type u} [CommRing R]
    {S : Type u} [CommRing S] [Algebra R S]
    [Module.FaithfullyFlat R S]
    {M : Type u} {N : Type u}
    [AddCommGroup M] [AddCommGroup N]
    [Module R M] [Module R N]
    (f : M →ₗ[R] N)
    (hf : UniversallyInjective (f.baseChange S)) :
    UniversallyInjective f
```

5. Mittag-Leffler 逆系统

本节的结果位于文件 lib/mlSystem.lean 中， [8]。我们使用 Mathlib 类型类 `InverseSystem` 对逆系统的经典 Mittag-Leffler 条件进行形式化。该类型类刻画了从预序集 ι 到类型范畴的逆变函子。具体而言，逆系统由一族类型 $(F_i)_{i\in\iota}$ 以及过渡映射$f_{ij} : F_j \to F_i$ 构成， $i \le j$并满足恒等与复合公理。按照 [1, tag0594], 的定义，我们称这样的系统为 *Mittag-Leffler* 系统，如果对每个 $i \in \iota$，像 $\mathrm{im}(f_{ij})$ 会随着 j 增大而稳定。在 Lean 中，我们将其形式化为:

```
class InverseSystem.IsMittagLeffler (F : ι→ Type*) ∀[ i, Nonempty (F i)]
(f : ∀⦃ i ⦄j, i ≤ j → F j → F i) [InverseSystem f] where
stabilization : ∀ i : ι, ∃ j : ι, ∃ (hij : i ≤ j), ∀k : ι,
 ∀ (hjk : j ≤ k), Set.range (f (hij.trans hjk)) = Set.range (f hij)
```

我们证明以下基础性结果：可数有向指标集上的 Mittag-Leffler 逆系统具有非空的逆极限:

定理 5.1. 设 ι 为可数有向预序，并设 (F_i, f_{ij}) 为以 ι为指标的 *Mittag-Leffler* 逆系统。则逆极限 $\varprojlim F_i$ 非空。

我们将其形式化为:

```
theorem nonempty_inverseLimit_of_countable_mittagLefflerι
    { : Type*} [Preorder ι] [IsDirected ι (· ≤ ·)] [Nonempty ι]
    [Countable ι]
    {F : ι→ Type*} ∀[ i, Nonempty (F i)]
    (f : ∀⦃ i ⦄j, i ≤ j → F j → F i) [InverseSystem f]
    [IsMittagLeffler F f] :
    Nonempty (InverseLimit f)
```

此处，我们采用如下逆极限的具体定义:

```
def InverseLimit ι{ : Type*} [Preorder ι] {F : ι→ Type*}
    (f : ∀⦃ i ⦄j, i ≤ j → F j → F i) : Type _ :=
  { x : ∀ i, F i // ∀ i j (hij : i ≤ j), f hij (x j) = x i }
```

需要指出的是，Mathlib 内置的 `InverseSystem.limit` 定义为对严格小于给定指标的元素取极限，这不适用于我们的目的，因此我们将上述定义作为一项新内容加入库中。

定理 5.1 的主要应用是以下关于逆系统短正合列的定理，它是将 Mittag-莱夫勒条件与下一节发展的模论联系起来的关键要素。给定可数有向指标集上的 $0 \to (A_i) \to (B_i) \to (C_i) \to 0$ 个 R-模构成的逆系统短正合列，若系统 (A_i) 满足 Mittag-莱夫勒条件，则诱导出的逆极限映射 $\varprojlim B_i \to \varprojlim C_i$ 是满射。更一般地说，有以下结论：

定理 5.2. 设 ι 为一个可数有向预序，并且

$$A_i \xrightarrow{f_i} B_i \xrightarrow{g_i} C_i \to 0$$

是由 R-模组成的逆系统的右正合列，即每个 g_i 都是满射，且 像$(f_i) =$ 核(g_i)。如果逆系统 (A_i) 满足 ***Mittag-Leffler*** 条件，则逆极限上的诱导映射

$$\varprojlim B_i \to \varprojlim C_i$$

是满射。

我们将其形式化为：

```
theorem surjective_limit_of_mittagLeffler_exactι
    { : Type*} [Preorder ι] [IsDirected ι (· ≤ ·)] [Nonempty ι]
    [Countable ι] {R : Type*} [Ring R]
    {A B C : ι→  Type*}∀
    [ i, AddCommGroup (A i)] ∀[ i, AddCommGroup (B i)]∀
    [ i, AddCommGroup (C i)] ∀[ i, Module R (A i)]∀
    [ i, Module R (B i)] ∀[ i, Module R (C i)]∀
    [ i, Nonempty (A i)] ∀[ i, Nonempty (B i)] ∀[ i, Nonempty (C i)]
    (fA : ∀⦃ i ⦄j, i ≤ j → A j →ₗ[R] A i)
    [InverseSystem (fun i j h => (fA h))]
    (fB : ∀⦃ i ⦄j, i ≤ j → B j →ₗ[R] B i)
    [InverseSystem (fun i j h => (fB h))]
    (fC : ∀⦃ i ⦄j, i ≤ j → C j →ₗ[R] C i)
    [InverseSystem (fun i j h => (fC h))]
    (f : ∀ i, A i →ₗ[R] B i)
    (g : ∀ i, B i →ₗ[R] C i)
    (hg_surj : ∀ i, Function.Surjective (g i))
    (hexact : ∀ i, LinearMap.range (f i) = LinearMap.ker (g i))
    (hf_compat : ∀ i j (h : i ≤ j) (a : A j), fB h (f j a) = f i (fA h a))
    (hg_compat : ∀ i j (h : i ≤ j) (b : B j), fC h (g j b) = g i (fB h b))
    [IsMittagLeffler A (fun i j h => (fA h))] :
    Function.Surjective (InverseLimit.map
      (fun i j h => (fB h))
      (fun i j h => (fC h))
      (fun i => (g i))
      (fun i j h => by
        ext x
        exact hg_compat i j h x))
```

6. 支配

本节的结果位于 [8]中的文件 lib/Domination.lean。Mittag-Leffler 模理论背后的一个关键技术概念是线性映射的支配。给定R-线性映射 $f : M \to N$ 和 $g : M \to M'$，我们称 g 支配 f，如果对于每个 R-模 Q,

$$\ker(f \otimes \mathrm{id}_Q) \subseteq \ker(g \otimes \mathrm{id}_Q).$$

在 Lean 中，这被形式化为:

```
def LinearMap.Dominates (f : M →ₗ[R] N) (g : M →ₗ[R] M') : Prop :=∀
(Q : Type u) (_ : AddCommGroup Q) (_ : Module R Q),∀
x : TensorProduct R M Q,(TensorProduct.map f LinearMap.id) x = 0 →
(TensorProduct.map g LinearMap.id) x = 0
```

请注意，在 Lean 代码中， `Dominates f g` 表示 g 支配 f。如果 f 和 g彼此支配，也就是说，每个映射都支配另一个映射，那么称它们彼此支配；即对于每个 R-模 Q，都有 $\ker(f\otimes \mathrm{id}_Q) = \ker(g\otimes \mathrm{id}_Q)$。
支配的一个基本刻画在整个 Mittag-Leffler 理论中都有应用，具体结果如下，对应于 [6,引理 6。10] 和 [1, 标签 059D]:

定理 6.1。 设 R 是一个交换环，且 $f : M \to N$ 和 $g : M \to M'$ 是R-线性映射。假设 $N/$像(f) 是有限表示的。那么， g 支配 f 当且仅当 g 经由 f分解，也就是说，存在一个 R-线性映射 $h : N \to M'$ 使得 $g = h \circ f$。

我们将其形式化如下:

```
theorem LinearMap.dominates_iff_factors_through (f : M →ₗ[R] N)
(g : M →ₗ[R] M') [Module.FinitePresentation R (N ⧸ LinearMap.range f)]:
Dominates f g ↔∃  (h : N →ₗ[R] M'), g = h.comp f
```

证明关键地运用了第 3 节的推出构造和第 4 节的普遍单射理论，通过以下连接这三种概念的中间结果:

定理 6.2。 设 R 是一个交换环，且 $f : M \to N$ 和 $g : M \to M'$ 是 R-线性映射。那么， g 支配 f，当且仅当典范映射 $\iota_C : M' \to$ 推出(f, g)是普遍单射的。

我们将其形式化如下:

```
theorem LinearMap.dominates_iff_pushout_inr_universallyInjective
    (f : M →ₗ[R] N) (g : M →ₗ[R] M') :
    Dominates f g ↔ UniversallyInjective (Pushout.inr f g)
```

7. MITTAG-LEFFLER 模

本节的结果位于 [8] 中的文件 lib/mlModule.lean。现在我们转而讨论 Mittag-Leffler 模的概念，该概念最早由 Raynaud 和 Gruson 在 [7] 中引入。我们继续在交换环 R 上工作。上一节中的支配概念在这里发挥着关键作用

。若对于每个有限表示的 R-模 P 以及每个 R-线性映射 $f : P \to M$，都存在一个有限表示的R-模 Q 和一个 R-线性映射 $g : P \to Q$，使得 f 和 g 彼此支配，则称 R-模 M 为 *Mittag-Leffler* 。在 Lean 中，这被形式化为：

```
def Module.IsMittagLeffler (R : Type u) [CommRing R]
(M : Type u) [AddCommGroup M] [Module R M] : Prop :=∀
(P : Type u) (_ : AddCommGroup P) (_ : Module R P)
(_ : Module.FinitePresentation R P),∀f : P →ₗ[R] M,∃
(Q : Type u) (_ : AddCommGroup Q) (_ : Module R Q)
(_ : Module.FinitePresentation R Q),∃g : P →ₗ[R] Q,
 LinearMap.MutuallyDominate f g
```

利用 `Module.isDirectLimit_of_finitelyPresented` 在第 2 节中所证明的结论，任意 R-模M都可写成有向余极限 $M = \varinjlim_{i \in \iota} M_i$，由有限表示的 R-模构成。根据 [6, 命题 6.11], 以及 [1, 标签 0599], Mittag-莱夫勒条件关于此类表示有若干等价的重新表述，我们现在加以说明。给定这样的表示，我们称其满足条件 (2)，如果对每个 $i \in \iota$，存在 $j \geq i$，使得典范映射 $M_i \to M$ 被过渡映射 $f_{ij} : M_i \to M_j$所支配。在 Lean 中：

```
def IsMittagLeffler' ι{ : Type u} [Preorder ι] [IsDirected ι (· ≤ ·)]
[Nonempty ι] [DecidableEq ι]
(F : ι→  Type u) ∀[ i, AddCommGroup (F i)] ∀[ i, Module R (F i)]∀
[ i, Module.FinitePresentation R (F i)]
(f : ∀ i j, i ≤ j → F i →ₗ[R] F j)
[DirectedSystem F (fun i j h => f i j h)] : Prop :=∀i : ι, ∃ j : ι,
 ∃ (hij : i ≤ j),
LinearMap.Dominates (Module.DirectLimit.of R ι F f i) (f i j hij)
```

条件 (3) 用显式分解取代支配：对于每个 i，存在 $j \geq i$，使得对于所有 $k \geq i$，映射 f_{ij} 经由 f_{ik} 分解。在 Lean 中：

```
def IsMittagLeffler'' ι{ : Type u} [Preorder ι] [IsDirected ι (· ≤ ·)]
[Nonempty ι]
(F : ι→  Type u) ∀[ i, AddCommGroup (F i)] ∀[ i, Module R (F i)]∀
[ i, Module.FinitePresentation R (F i)]
(f : ∀ i j, i ≤ j → F i →ₗ[R] F j) : Prop :=∀i : ι, ∃ j : ι,
 ∃ (hij : i ≤ j), ∀ k : ι, ∀ (hik : i ≤ k),∃(h : F k →ₗ[R] F j),
 f i j hij = h.comp (f i k hik)
```

最后，条件（4）将模论定义与第5节的逆系统理论联系起来。对于固定的 R-模 N，族 (同态集$(M_i, N))_{i\in\iota}$（其转移映射由与 f_{ij} 前复合给出）构成一个逆系统；条件（4）要求对于每个 N，该逆系统都满足 Mittag-Leffler 意义下的 `InverseSystem.IsMittagLeffler`。

我们将依照 [6, 命题 6。11], 和[1, 标签 0599]: ， 对这些条件的等价性进行形式化。

定理 7.1。 设 R 是一个交换环， ι 是一个有向指标集，且 $M = \varinjlim_{i\in\iota} M_i$ 是 R-模 M 的一个表示，其中 M 是有限表示模的有向余极限。则以下条件等价:

(1) M 满足 *Mittag-Leffler* 条件。*(2)* 该表示满足条件 (2)。*(3)* 该表示满足条件 (3)。*(4)* 对每个 R-模 N，逆系统 (同态集(M_i, N)) 都满足 *Mittag-Leffler* 条件。

我们注意到， (1) 与 (2) 的等价性需要比[6, 命题 6.11], 和 [1, 标签 0599]中所述更加谨慎地处理：其中给出的一个方向的论证，在处理直接极限的转移映射时不够严谨，因此我们给出修正后的证明。我们将这些等价关系形式化为三个独立的双条件命题:

```
theorem isMittagLeffler_iff_isMittagLeffler' :
    IsMittagLeffler' R F f ↔ IsMittagLeffler R (Module.DirectLimit F f)

theorem isMittagLeffler'_iff_isMittagLeffler'' :
    IsMittagLeffler' R F f ↔ IsMittagLeffler'' R F f

theorem isMittagLeffler''_iff_isMittagLeffler'''
    (hf : ∀ i, f i i (le_refl i) = LinearMap.id)
    (hcomp : ∀ i j k (hij : i ≤ j) (hjk : j ≤ k),
      f i k (hij.trans hjk) = (f j k hjk).comp (f i j hij)) :
    IsMittagLeffler'' R F f ↔ IsMittagLeffler''' R F f hf hcomp
```

我们还证明了以下基本保持性质。有限表示模是 Mittag-Leffler 模：

```
theorem Module.finitePresentation_isMittagLeffler (M : Type u)
    [AddCommGroup M] [Module R M] [Module.FinitePresentation R M] :
    IsMittagLeffler R M
```

并且投射模是 Mittag-Leffler 模：

```
theorem Module.proj_is_Mittag_Leffler {R : Type u} {P : Type u} [CommRing
     R]
    [AddCommGroup P] [Module R P] [Module.Projective R P] :
    IsMittagLeffler R P
```

我们对后一个结果的证明与 [1, 6]中给出的证明不同，也更简单。他们利用了 Mittag-Leffler 性质的另一种刻画，该刻画涉及这些模在与直积作张量积时的表现。我们通过直接而简单地证明以下事实，避开了这种复杂的形式化：自由模是 Mittag-Leffler 模，而 Mittag-Leffler 模的直和项也是 Mittag-Leffler 模。这两点都可直接由 Mittag-Leffler 模的定义推出。

本节的主要结构性结果是下面的定理，它是证明定理II的关键要素，对应于 [6, 引理 7。1] 以及 [1, 标签 059W]:

定理 7.2。 设 R 是一个交换环，设 $M = \varinjlim_{i \in \iota} M_i$ 是 M 作为有限表示模的有向余极限的一个表示，并假设 M 既是 *Mittag-Leffler* 的，又是可数生成的。那么存在一个可数有向子集 $I' \subseteq \iota$，使得$M \cong \varinjlim_{i \in I'} M_i$。

在 Lean 中，这被形式化为：

```
theorem mittagLeffler_countablyGenerated_has_countable_colimitι
{ : Type u} [Preorder ι] [IsDirected ι (· ≤ ·)] [Nonempty ι]
[DecidableEq ι]
(F : ι→  Type u) ∀[ i, AddCommGroup (F i)] ∀[ i, Module R (F i)]∀
[ i, Module.FinitePresentation R (F i)](f : ∀ i j, i ≤ j → F i →ₗ[R] F j)
[DirectedSystem F (fun i j h => f i j h)]
(hML : IsMittagLeffler R (Module.DirectLimit F f))
(hCountGen : Module.IsCountablyGenerated R (Module.DirectLimit F f)) ↹∃
(I' : Set ι) (_ : I'.Countable) (_ : DirectedOn (· ≤ ·) I')
(_ : I'.Nonempty),Nonempty (Module.DirectLimit F f ≃ₗ[R]
Module.DirectLimit (fun i : I' => F i.val)
(fun i j (hij : i ≤ j) => f i.val j.val hij))
```

证明分为两个主要步骤。首先，利用可数生成假设，可以找到 M 的一个可数生成元集合，其中每个生成元都由 ι 中某个指标处的元素表示，从而得到一个初始可数子集 $I_0 \subseteq \iota$。其次，逐步扩充 I_0：在每一步中，对当前集合中的每个指标 i，Mittag-莱夫勒条件都会给出一个稳定化指标 $j_i \geq i$，并将其加入集合。取这一序列的并，就得到所需的可数有向子集 I'；而 $M \cong \varinjlim_{i \in I'} M_i$ 的成立，则源于 I' 的构造使其包含了 M 的所有生成元的代表元。其中还包含一个我们证明的独立引理，`lemma countable_subset_directed_closure`：若从有向偏序集中的一个可数子集出发，则可以将其扩充为一个包含原集合的可数有向子集。

8. 投射模与忠实平坦映射

本节的结果位于文件 basechange.lean 中，即 [8]。我们现在已具备证明两个主定理所需的全部要素。我们从投射模的刻画开始，即定理 II，并将其形式化为：

```
theorem proj_iff_Mittag_Leffler {R : Type u} {P : Type u} [CommRing R]
    [AddCommGroup P] [Module R P] : Module.Projective R P ↔
    (Module.Flat R P) ∧ (Module.IsMittagLeffler R P) ∧
    (IsDirectSumOfCountablyGenerated R P)
```

正向的证明是显然的：根据 Mathlib 中已有的标准结果，投射性蕴含平坦性；根据 `proj_is_Mittag_Leffler`，它还蕴含 Mittag-Leffler 性质；而根据第 1 节中的 `proj_is_dirSumCountable`，它还蕴含直和分解。对于反向，假设 $P = \bigoplus_{i \in I} M_i$ 是可数生成模的直和，其中每个模

作为 P 的直和项，继承了平坦性和 Mittag-Leffler 性质。因此，只需证明每个 M_i 都是投射的。这将问题归结为以下关键结果，它处理了可数生成的情形：

```
theorem proj_if_countable_flat_Mittag_Leffler {R : Type u} {P : Type u}
    [CommRing R] [AddCommGroup P] [Module R P]
    (PFlat : Module.Flat R P) (PML : Module.IsMittagLeffler R P)
    (PCount : ∃ s : Set P, s.Countable ∧ span R s = ⊤) :
    Module.Projective R P
```

证明应用 Lazard 定理，将 P 写成有限表示自由模的有向余极限，然后利用第 7 节中的 `mittagLeffler_countablyGenerated_has_countable_colimit`将指标集替换为可数指标集。接下来，关键步骤是通过提升性质应用投射性判据：给定一个满射 $f : N_2 \to N_3$ 和一个映射 $g : P \to N_3$，我们需要将 g 提升经过 f。将 $P = \varinjlim_{i \in I'} G_i$ 写成关于可数有向指标集的逆极限后，映射 $g \circ \iota_i : G_i \to N_3$ 组装成逆极限 $\varprojlim$ 同态集 (G_i, N_3) 中的一个元素。由于 P 是 Mittag-Leffler 的，对于任意（同态集(G_i, N)），逆系统都是 Mittag-Leffler 的；这是由定理 7.1中条件（1）与（4）的等价性得出的。接着，满射定理（定理 5.2）给出一个相容的提升族，属于

$$\varprojlim \mathrm{Hom}(G_i, N_2),$$

它通过 `DirectLimit.lift` 组装成所需的映射 $P \to N_2$。

现在我们转向主定理。在组装定理 I 的证明之前，需要两个关键的下降引理。第一个引理指出，忠实平坦性能够反映 Mittag-Leffler 性质。这是对 [6, 引理 9.2], 的改进，该引理已见于 [1, 标签 05A5], ，其证明归功于 Juan Pablo Acosta Lopez。

定理 8.1。 设 $R \to S$ 是交换环之间的忠实平坦映射，且 M 是一个R-模。如果 $S \otimes_R M$ 在 S上是 *Mittag-Leffler* 的，那么 M 在 R上也是 *Mittag-Leffler* 的。

我们将其形式化为：

```
theorem ML_of_faithfullyFlat_tensorProduct {R : Type u} {S : Type u}
    {M : Type u} [CommRing R] [CommRing S] [Algebra R S]
    [Module.FaithfullyFlat R S] [AddCommGroup M] [Module R M]
    (h : Module.IsMittagLeffler S (TensorProduct R S M)) :
    Module.IsMittagLeffler R M
```

证明将 M 写成有限表示模的有向余极限 $\varinjlim G_i$，并利用定理 7.1中条件 (1) 和 (2) 的等价性。假设为基变换系统 i 中的每个 $(S \otimes_R G_i)$都给出了一个稳定化指标，关键步骤是结合支配关系的推出刻画定理 6.2 以及 `UniversallyInjective.of_baseChange_faithfullyFlat` 第 4 节中的内容，将其转移到原系统。

第二下降引理指出，忠实平坦性可以反映可数生成性 [1, 标签 0GVD]:

定理 8.2。 设 $R \to S$ 是交换环之间的忠实平坦映射，且 P 是一个 R-模。若 $S \otimes_R P$ 在 S 上可数生成，则 P 在 R 上可数生成。

我们将其形式化为:

```
theorem countably_generated_faithfully_flat {R : Type u} {S : Type u}
{P : Type u} [CommRing R] [CommRing S] [Algebra R S]
[Module.FaithfullyFlat R S] [AddCommGroup P] [Module R P]
(PSIsCountable : ∃ s : Set (TensorProduct R S P),
s.Countable ∧ span S s = ⊤) :∃t : Set P, t.Countable ∧ span R t = ⊤
```

初等证明先枚举 $S \otimes_R P$ 的一个可数生成集，将每个生成元表示为有限个纯张量之和，再把所有 P 分量收集成一个可数子集 $t \subseteq P$，然后利用忠实平坦性证明 t 在 R 上生成 P。
将这两个下降引理与 `proj_if_countable_flat_Mittag_Leffler` 结合，立即得到定理 I 的可数生成情形:

```
theorem proj_countable_faithfully_flat {R : Type u} {S : Type u}
    {P : Type u} [CommRing R] [CommRing S] [Algebra R S]
    [Module.FaithfullyFlat R S] [AddCommGroup P] [Module R P]
    (PIsCountable : ∃ s : Set (TensorProduct R S P),
      s.Countable ∧ span S s = ⊤)
    (baseChangeProj : Module.Projective S (TensorProduct R S P)) :
    Module.Projective R P
```

定理 I 的一般情形需要通过超限归纳论证化归到可数生成的情形。关键思路是构造一个由序数指标化、指标延伸至 $\gamma = \mathrm{succ}(\mathrm{type}(P))$ 的 Kaplansky 型滤过 $0 = P_0 \subseteq P_1 \subseteq \cdots \subseteq P_\gamma = P$，使得每个连续商 $P_{\alpha+1}/P_\alpha$ 都是 R 上可数生成的投射模。这个滤过通过超限递归构造：在每个后继步骤中，利用 P 的良序选取尚未属于 P_α 的最小元素，并由 `countably_generated_lift` 以一个可数生成的部分扩展当前阶段，同时保持 R-子模与分解 $S \otimes_R P = \bigoplus_{i \in I} Q_i$ 的子直和之间的对应关系。在极限阶段，取并。利用 `proj_countable_faithfully_flat` 证明每个连续商都是投射模；这依赖于同构 $S \otimes_R (P_{\alpha+1}/P_\alpha) \cong (\bigoplus_{i \in I'} Q_i)/(\bigoplus_{i \in I} Q_i)$，其中指标集 $I \subseteq I'$ 取适当值，而该同构本身是 $S \otimes_R P$ 的直和项，因此是投射模。由此，该滤过构成一个 Kaplansky 逐层分解

对于 P，根据 `kaplansky_devissage_iff_direct_sum` 可将 P 表示为可数生成投射模的直和，因此它是投射模。由此我们得到定理 I：

```
theorem proj_faithfully_flat {R : Type u} {S : Type u} {P : Type u}
    [CommRing R] [CommRing S] [Algebra R S] [Module.FaithfullyFlat R S]
    [AddCommGroup P] [Module R P] :
    Module.Projective R P ↔ Module.Projective S (TensorProduct R S P)
```

参考文献

1. Stacks Project 作者，*Stacks* 项目，https://stacks.math.columbia.edu/。2. Hyman Bass，诺特环上的内射维数，《美国数学会汇刊》**102** (1962)，18–29。MR 138644 3. L. Gruson，诺特交换环上平坦模的同调维数，《数学研讨会论文集》第 XI 卷（INDAM，罗马，1971 年交换代数会议及 INDAM，罗马，1972 年几何学会议），Academic Press，伦敦—纽约，1973 年，第 243–254 页。MR 337935 4. Irving Kaplansky，投射模，《数学年刊》（第二辑）**68** (1958)，372–377。MR 100017 5. Daniel Lazard，关于平坦模，《巴黎科学院院刊》**258** (1964)，6313–6316。MR 168625 6. Alexander Perry，模的投射性之忠实平坦下降，2010，arxiv:1011.0038v1。7. Michel Raynaud 和 Laurent Gruson，平坦性与投射性判据：模“平坦化”技术，《数学发明》**13** (1971)，1–89。MR 308104 8. Liran Shaul，*Lean* 中忠实平坦下降射影性，https://github.com/LiranShaul/lean-faithfully-flat-descent，2026 年，提交版本 80ff3db。

代数系，布拉格查理大学数学与物理学院，Sokolovská 83，186 75 布拉格，捷克共和国电子邮件地址：shaul@karlin.mff.cuni.cz