# Flat Modules

Jujian Zhang

# $- \otimes M$ is always right exact

$$0 \longrightarrow A \xrightarrow{f} B \xrightarrow{g} C \longrightarrow 0$$

$$\color{red}{0 \longrightarrow} A \otimes_R M \xrightarrow{f \otimes 1} B \otimes_R M \xrightarrow{g \otimes 1} C \otimes_R M \longrightarrow 0$$

# ...but not always left exact

For example $\mathbb{Z}/(2)$ as an $\mathbb{Z}$-module

$$0 \longrightarrow \mathbb{Z} \xrightarrow{\times 2} \mathbb{Z} \longrightarrow \mathbb{Z}/(2) \longrightarrow 0$$

gives

$$\begin{tikzcd}
0 \arrow[r, rightsquigarrow] & \mathbb{Z} \otimes_{\mathbb{Z}} \mathbb{Z}/(2) \arrow[r] \arrow[d, "\wr"', no head] & \mathbb{Z} \otimes_{\mathbb{Z}} \mathbb{Z}/(2) \arrow[r] \arrow[d, "\wr"', no head] & \mathbb{Z} \otimes_{\mathbb{Z}} \mathbb{Z}/(2) \arrow[r] \arrow[d, "\wr"', no head] & 0 \\
0 \arrow[r, rightsquigarrow] & \mathbb{Z}/(2) \arrow[r, "0"] & \mathbb{Z}/(2) \arrow[r] & \mathbb{Z}/(2) \arrow[r] & 0
\end{tikzcd}$$

# When is tensoring exact?

| | | |
|---|---|---|
| $- \otimes_R M$ is exact | `protected def ses : Prop :=`<br>`(tensor_right (Module.of R M)).is_exact` | (1) |
| $- \otimes M$ preserves mono | `protected def inj : Prop :=`<br>`∀ {⦃N N' : Module.{u} R⦄} (L : N ⟶ N'),`<br>`  function.injective L →`<br>`  function.injective`<br>`    ((tensor_right (Module.of R M)).map L)` | (2) |
| $I \otimes_R M \to R \otimes_R M$ is injective for all ideals $I$ | `protected def ideal : Prop :=`<br>`∀ (I : ideal R),`<br>`  function.injective (tensor_embedding M I)` | (3) |
| $I \otimes_R M \to R \otimes_R M$ is injective for all finitely generated ideals $I$ | `protected def fg_ideal : Prop :=`<br>`∀ (I : ideal R), I.fg →`<br>`  function.injective (tensor_embedding M I)` | (4) |

# (4) implies (3)

$$\begin{array}{c}(1)\\ \Updownarrow\\ (2)\\ \Downarrow\\ (3)\\ \Downarrow\\ (4)\end{array}$$

Fix an arbitrary ideal $I$. The set of all finitely generated subideals of $I$ is directed with respect to $\leq$, i.e. for any two finitely generated subideals $J, J'$, there is another finitely generated subideals larger than both, namely $J \sqcup J'$.

```
@[ext]
structure fg_subideal :=
(to_ideal : ideal R)
(fg : to_ideal.fg)
(le : to_ideal ≤ I)
```

```
instance : is_directed
    (fg_subideal I) (≤) :=
{ directed := λ J J',
  ⟨⟨J.to_ideal ⊔ J'.to_ideal,
    submodule.fg.sup J.fg
    J'.fg,
    sorry⟩, sorry⟩ }
```

# (4) implies (3)

## Lemma (ideal as colimit of finitely generated subideals)

(1)

$$I \cong \operatorname*{colim}_{J \leq I} J$$

$\Updownarrow$

*where $J$ runs over all finitely generated subideals.*

(2)

### Proof.

$\Downarrow$

By using universal property of colimit over directed system, $\operatorname{colim}_J J \to I$ can be realised by lifting the obvious linear map $J \to I$. For the other direction, if $i \in I$, then the principal subideal $\langle i \rangle$ is finitely generated, so there is a map $\langle i \rangle \to \operatorname{colim}_J J$. □

(3)

$\Downarrow$

(4)

### Corollary

$$I \otimes M \cong (\operatorname{colim}_{J \leq I} J) \otimes_R M$$

# (4) implies (3)

$$(1) \Updownarrow (2) \Downarrow (3) \Downarrow (4)$$

```
def as_direct_limit :=
module.direct_limit (λ (i : fg_subideal I), i.to_ideal) $
  λ i j hij, (submodule.of_le hij : i.to_ideal →ₗ[R] j.to_ideal)

def from_as_direct_limit :
  I.as_direct_limit →ₗ[R] I :=
module.direct_limit.lift R _ _ _ (λ i, submodule.of_le i.le) $
  λ i j hij r, rfl

@[simps]
def to_as_direct_limit :
  I →ₗ[R] I.as_direct_limit :=
{ to_fun := λ r,
    module.direct_limit.of R (fg_subideal I) (λ i, i.to_ideal)
      (λ _ _ h, submodule.of_le h) (principal_fg_subideal r)
      ⟨r, mem_principal_fg_subideal r⟩,
  ..sorry }
```

# (4) implies (3)

$$(1) \Updownarrow (2) \Downarrow (3) \Downarrow (4)$$

## Lemma

*colimits over direct system commutes with tensor.*

## Proof.

Consider $(\mathrm{colim}_{i\in\mathcal{I}}\, i) \otimes_R M$ and $\mathrm{colim}_{i\in\mathcal{I}}\,(i \otimes_R M)$. The forward direction is done by using universal property of tensor product, we construct a bilinear map $(\mathrm{colim}_{i\in\mathcal{I}}\, i) \to M \to \mathrm{colim}_{i\in\mathcal{I}}\,(i \otimes_R M)$: for each $i \in \mathcal{I}$ and $x \in i$, there is a map $M \xrightarrow{-\otimes x} M \otimes i \longrightarrow \mathrm{colim}_{i\in\mathcal{I}}\,(i \otimes_R M)$. The other direction is by descending the family of maps $i \otimes_R M \to (\mathrm{colim}_{i\in\mathcal{I}}\, i) \otimes_R M$ for all $i \in \mathcal{I}$. $\square$

## Corollary

$$I \otimes M \cong \mathrm{colim}_{J\leq I}\,(J \otimes_R M)\,.$$

# (4) implies (3)

$$\begin{array}{c}(1)\\ \Updownarrow\\ (2)\\ \Downarrow\\ (3)\\ \Downarrow\\ (4)\end{array}$$

```
def
    direct_limit_of_tensor_product_to_tensor_product_with_direct_limit
    :
  direct_limit (λ i, G i ⊗[R] M)
    (λ i j hij, tensor_product.map (f _ _ hij) (linear_map.id)) →ₗ
    [R] (direct_limit G f) ⊗[R] M :=
direct_limit.lift _ _ _ _ (λ i, tensor_product.map (direct_limit.of
    _ _ _ _ _) linear_map.id) $
λ i j hij z, sorry
```

# (4) implies (3)

$$(1) \Updownarrow (2) \Downarrow (3) \Downarrow (4)$$

```
def
    tensor_product_with_direct_limit_to_direct_limit_of_tensor_product
    :
(direct_limit G f) ⊗[R] M →ₗ[R] direct_limit (λ i, G i ⊗[R] M)
  (λ i j hij, tensor_product.map (f _ _ hij) (linear_map.id)) :=
tensor_product.lift $ direct_limit.lift _ _ _ _ (λ i,
{ to_fun := λ g,
  { to_fun := λ m, direct_limit.of R ι (λ i, G i ⊗[R] M)
        (λ i j hij, tensor_product.map (f _ _ hij) (linear_map.id))
    i $ g ⊗ₜ m,
    ..sorry }) ..sorry }
```

# (4) implies (3)

$$\begin{array}{c} (1) \\ \Updownarrow \\ (2) \\ \Downarrow \\ (3) \\ \Updownarrow \\ (4) \end{array}$$

It is a calculation away to show that

$$\begin{tikzcd} I \otimes_R M \arrow[r] \arrow[d, dash, "\cong"] & R \otimes_R M \\ \operatorname{colim}_{J \leq I} (J \otimes_R M) \arrow[ur, "\bar{\iota}"] & \end{tikzcd}$$

where $\bar{\iota}$ is obtained via the family of maps $J \otimes_R M \to R \otimes_R M$. Since each $J \otimes_R M \to R \otimes_R M$ is injective, $\bar{\iota}$ is injective as well, hence $I \otimes_R M \to R \otimes_R M$ is injective.

# (3) implies (2)

(1)
$\Updownarrow$
(2)
$\Downarrow$
(3)
$\Updownarrow$
(4)

We define the character module of $M$ to be $M^* := \mathrm{Hom}_{\mathbb{Z}}(M, \mathbb{Q}/\mathbb{Z})$. $M^*$ is an $R$-module by $(r \bullet f)(m) := f(r \bullet m)$. Let $L : N \to N'$, then $L^* : N'^* \to N^*$ defined by $L^* := - \circ L$ makes taking character module a contravariant functor.

**Theorem**

*If $L$ is a monomorphism then $L^*$ is an epimorphism.*

**Proof.**

Let $g \in N^* = \mathrm{Hom}_{\mathbb{Z}}(N, \mathbb{Q}/\mathbb{Z})$, since $\mathbb{Q}/\mathbb{Z}$ is an injective group, $g : N \to \mathbb{Q}/\mathbb{Z}$ factors as $N \xrightarrow{L} N' \xrightarrow{\bar{g}} \mathbb{Q}/\mathbb{Z}$ so that $L^*(\bar{g}) = g$. $\square$

# (3) implies (2)

$$(1) \Updownarrow (2) \Downarrow (3) \Updownarrow (4)$$

```
@[derive add_comm_group]
def rat_circle : Type* :=
  ulift $ ℚ ⧸ (algebra_map ℤ ℚ).to_add_monoid_hom.range
```

```
@[derive add_comm_group]
def character_module : Type w :=
M →ₗ[ℤ] rat_circle.{w}
```

```
instance character_module.module : module R (character_module M) :=
{ smul := λ r f, { to_fun := λ m, f (r • m), ..sorry } ..sorry }
```

# (3) implies (2)

$$\begin{array}{c}(1)\\ \Updownarrow\\ (2)\\ \Downarrow\\ (3)\\ \Updownarrow\\ (4)\end{array}$$

### Theorem (injective cogenerator)

*For every $M \ni m \neq 0$, there is some $h \in M^*$ such that $h(m) \neq 0$.*

### Proof.

It is sufficient to construct a map $h' : \langle m \rangle \to \mathbb{Q}/\mathbb{Z}$ such that $h'(m) \neq 0$ by injectivity of $\mathbb{Q}/\mathbb{Z}$. Either the additive order of $m$ is finite or infinite:

- If $m$ has finite order $n$, then $h'$ is defined as $k \bullet m \mapsto \frac{k}{n} \mod 1$. This is well-defined because if $k \bullet m = k' \bullet m$ then $k - k' \mid n$ thus $\frac{k}{n} \equiv \frac{k'}{n} \mod 1$.
- If $m$ has infinite order, then $h'$ is defined as $k \bullet m \mapsto \frac{k}{37}$. In this case, it is well defined because the choice of $k$ is unique.

$\square$

# (3) implies (2)

$$(1) \Updownarrow (2) \Downarrow (3) \Updownarrow (4)$$

```
lemma non_zero {m : M} (hm : m ≠ 0) : ∃ (h : character_module M),
    h m ≠ 0 :=
begin
  let M' : submodule ℤ M := submodule.span ℤ {m},
  suffices : ∃ (h' : M' →ₗ[ℤ] rat_circle), h' ⟨m,
    submodule.subset_span (set.mem_singleton _)⟩ ≠ 0,
  { sorry },
  by_cases h_order : add_order_of m ≠ 0,
  { sorry },
  { sorry },
end
```

# (3) implies (2)

$$\begin{array}{c}(1)\\ \Updownarrow\\ (2)\\ \Downarrow\\ (3)\\ \Updownarrow\\ (4)\end{array}$$

### Theorem (Tensor-Hom adjunction)

*Let $R, S$ be two commutative rings and $X$ and $R, S$-bimodule, then $- \otimes_R X \dashv \mathrm{Hom}_S(X, -)$, where for $R$-module $Y$, $Y \otimes_R X$ has the $S$-module structure given by $s \bullet (y \otimes x) := y \otimes (s \bullet x)$ and for $S$-module $Z$, $\mathrm{Hom}_S(X, Z)$ has the $R$-module structure given by $r \bullet l := x \mapsto l(r \bullet x)$.*

### Proof.

Let $Y$ be an $R$-module and $Z$ an $S$-module. Any $l : Y \otimes X \to Z$ also gives a map $Y \to \mathrm{Hom}_S(X, Z)$ by $y \mapsto x \mapsto l(y \otimes x)$. Conversely, for any $R$-linear map $l : Y \to \mathrm{Hom}_S(X, Z)$, $y \otimes x \mapsto l(y)(x)$ defines an $S$-linear map. □

### Corollary

$$\mathrm{Hom}_R(N, M^*) \cong (N \otimes_R M)^*$$

# (3) implies (2)

$$\begin{array}{c}(1)\\ \Updownarrow \\ (2)\\ \Downarrow \\ (3)\\ \Updownarrow \\ (4)\end{array}$$

```
@[simps]
def hom_equiv.inv_fun' {Y : Module.{v} R'} {Z : Module.{v} S'} (l :
    Y →ₗ[R'] (X' →ₗ[S'] Z)) :
  ((tensor_functor R' S' X').obj Y ⟶ Z) :=
{ to_fun := (add_con_gen _).lift (free_add_monoid.lift $ show Y ×
    X' → Z, from λ p, l p.1 p.2) $
    add_con.add_con_gen_le $ λ p p' (h : eqv R' Y X' p p'), _,
  ..sorry }

def tensor_product.lift {R : Type u_1}  [comm_semiring R]
  {M : Type u_4}  {N : Type u_5} {P : Type u_6}
  [add_comm_monoid M]  [add_comm_monoid N]
  [add_comm_monoid P]  [module R M]
  [module R N]  [module R P]
  (f : M →ₗ[R] N →ₗ[R] P) : tensor_product R M N →ₗ[R] P
```

# (3) implies (2)

$$(1) \Updownarrow (2) \Downarrow (3) \Updownarrow (4)$$

**Theorem (Lambek)**

*If $M^*$ is injective, then $M$ is flat.*

**Proof.**

Let $A, B$ be $R$-modules and $L : A \to B$ an injective $R$-linear map. If $A \otimes_R M \ni z \neq 0$ is in the kernel of $L \otimes 1$, then there would be some $g \in (A \otimes M)^*$ such that $g(z) \neq 0$. Let $f : A \to M^*$ be defined as $a \mapsto m \mapsto g(a \otimes m)$, since $M^*$ is injective and $L$ is mono, $f$ factors through $f' : B \to M^*$, let $g' \in (B \otimes_R M)^*$ be the corresponding map of $f'$ under bijection $\mathrm{Hom}_R(B, M^*) \cong (B \otimes_R M)^*$. By writing $z$ as $\sum_i a_i \otimes m_i$, since $(L \otimes 1)(z) = 0$, we derive $g'(\sum_i L(a_i) \otimes m_i) = g'(0) = 0$. $\square$

# (3) implies (2)

$$\begin{array}{c}(1)\\ \Updownarrow\\ (2)\\ \Downarrow\\ (3)\\ \Updownarrow\\ (4)\end{array}$$

### Theorem (Lambek)

*If $M^*$ is injective, then $M$ is flat.*

### Proof.

.

$$\begin{aligned} g'\left(\sum_i L(a_i)\otimes m_i\right) &= \sum_i g'(L(a_i)\otimes m_i) = \sum_i f'(L(a_i))(m_i) \\ &= \sum_i f(a_i)(m_i) = \sum_i g(a_i\otimes m_i) \\ &= g\left(\sum_i a_i\otimes m_i\right) = g(z)\neq 0. \end{aligned}$$

This is contradiction, thus $z$ must be zero which means $\ker(L\otimes 1)=0$. □

# (3) implies (2)

$$\begin{array}{c}(1)\\ \Updownarrow\\ (2)\\ \Updownarrow\\ (3)\\ \Updownarrow\\ (4)\end{array}$$

## Corollary

*If $\iota : I \otimes_R M \to R \otimes_R M$ is injective for all ideals $I$, then $M$ is flat.*

## Proof.

By Baer's criterion, it is sufficient to show the restriction map $\mathrm{Hom}_R(R, M^*) \to \mathrm{Hom}_R(I, M^*)$ is surjective for all ideals $I$. Fix an ideal $I$, let $f : I \to M^*$ corresponding to $f' \in (I \otimes_R M)^*$. Since $\iota$ is injective, $\bar{\iota}$ is surjective so that there is an $F \in (R \otimes_R M)^*$ such that $\bar{\iota}(F) = f'$. Then $F$ induces the required $\mathrm{Hom}_R(R, M^*)$. □

# In terms of Tor functor

| | | |
|---|---|---|
| $\mathrm{Tor}_i^R(N, M) \cong 0$ for all $R$-modules $N$ and $1 \leq i$ | `∀ (n : ℕ) (hn : 0 < n) (N : Module.{u} R), nonempty (((Tor' (Module.{u} R) n).obj N).obj M ≅ 0)` | (5) |
| $\mathrm{Tor}_1^R(N, M) \cong 0$ for all $R$-modules $N$ | `∀ (N : Module.{u} R), nonempty (((Tor' (Module.{u} R) 1).obj N).obj M ≅ 0)` | (6) |
| $\mathrm{Tor}_1^R(R/I, M) \cong 0$ for all ideals $I$ | `∀ (I : ideal R), nonempty (((Tor' (Module.{u} R) 1).obj (Module.of R (R / I))).obj M ≅ 0)` | (6) |
| $\mathrm{Tor}_1^R(R/I, M) \cong 0$ for all finitely generated ideals $I$ | `∀ (I : ideal R), nonempty (((Tor' (Module.{u} R) 1).obj (Module.of R (R / I))).obj M ≅ 0)` | (8) |

# flatness implies (5)

$$\begin{array}{c} \text{flat} \\ \Downarrow \\ (5) \\ \Downarrow \\ (6) \\ \Downarrow \\ (7) \\ \Downarrow \\ (8) \end{array}$$

**Theorem**

*The category of $R$-modules has enough free objects.*

**Proof.**

Note that $\bigoplus_M R$ is a free module. There is a canonical surjective map $\bigoplus_M R \to M$ whose $M \ni m$-th coordinate □

# flatness implies (5)

$$\begin{array}{c} \text{flat} \\ \Downarrow \\ (5) \\ \Downarrow \\ (6) \\ \Downarrow \\ (7) \\ \Downarrow \\ (8) \end{array}$$

## Theorem

*The category of R-modules has enough free objects.*

```
def afree : Module.{u} R := Module.of R $ ⨁ (m : M), R

def from_afree : M.afree ⟶ M :=
direct_sum.to_module _ _ _ $ λ m, { to_fun := λ r, r · m ..sorry}
```

# flatness implies (5)

flat

$\Downarrow$

(5)

$\Downarrow$

(6)

$\Downarrow$

(7)

$\Downarrow$

(8)

**Theorem**

*The category of $R$-modules has enough free objects.*

**Corollary**

*Continuing inductively, the first row is a free resolution.*

$$\begin{tikzcd}
\cdots \arrow[r] & \bigoplus_{\ker\epsilon} R \arrow[rr] \arrow[dr, two heads] & & \bigoplus_M R \arrow[r, "\epsilon"] & M \\
& & \ker\epsilon \arrow[ur, hook] & &
\end{tikzcd}$$

Using this resolution and the fact that $M$ is flat, we see that all higher Tor vanishes.

# flatness implies (5)

flat
⇓
(5)
⇓
(6)
⇓
(7)
⇓
(8)

```
def free_res.chain_complex.Xd_aux :
ℕ →
Σ' (N_prev N_next : Module.{u} R)
  (h : module.free R N_prev ∧ module.free R N_next),
  N_next ⟶ N_prev :=
@nat.rec
  (λ _, Σ' (N_prev N_next : Module.{u} R)
    (h : module.free R N_prev ∧ module.free R N_next),
    N_next ⟶ N_prev)
⟨M.afree, (kernel M.from_afree).afree,
  ⟨Module.afree_is_free _, Module.afree_is_free _⟩,
  Module.from_afree _ ≫ kernel.ι _⟩ $ λ n P,
⟨P.2.1, (kernel P.2.2.2).afree, ⟨P.2.2.1.2, Module.afree_is_free _⟩,
    Module.from_afree _ ≫ kernel.ι _⟩
```

# (8) implies flatness

$$\begin{tikzcd}
\text{flat} \arrow[d, Rightarrow] \\
(5) \arrow[d, Rightarrow] \\
(6) \arrow[d, Rightarrow] \\
(7) \arrow[d, Rightarrow] \\
(8) \arrow[uuuu, dashed, Rightarrow, bend left]
\end{tikzcd}$$

Let $I$ be any finitely generated ideal, consider the exact sequence

$$0 \longrightarrow I \longrightarrow R \longrightarrow R/I \longrightarrow 0,$$

```
def ses_of_ideal (I : ideal R) : short_exact_sequence (Module.{u}
    R) :=
{ fst := Module.of R I,
  snd := Module.of R R,
  trd := Module.of R (R / I),
  f := Module.of_hom ⟨coe, λ _ _, rfl, λ _ _, rfl⟩,
  g := submodule.mkq I,
  mono' := sorry,
  epi' := sorry,
  exact' := sorry }
```

# (8) implies flatness

$$\begin{tikzcd} \text{flat} \arrow[d, Rightarrow] \\ (5) \arrow[d, Rightarrow] \\ (6) \arrow[d, Rightarrow] \\ (7) \arrow[d, Rightarrow] \\ (8) \arrow[uuuu, Rightarrow, dashed, bend left] \end{tikzcd}$$

Let $I$ be any finitely generated ideal, consider the exact sequence

$$0 \longrightarrow I \longrightarrow R \longrightarrow R/I \longrightarrow 0,$$

it induces

$$\cdots \longrightarrow \mathrm{Tor}_1^R(R/I, M) \xrightarrow{\delta} I \otimes_R M \longrightarrow R \otimes_R M \longrightarrow R/I \otimes_R M.$$

```
def δ₀ (A : short_exact_sequence C) := δ F 0 A ≫
    (left_derived_zero_iso_self F).hom.app A.1

lemma seven_term_exact_seq (A : short_exact_sequence C) :
  exact_seq D [
    (F.left_derived 1).map A.f, (F.left_derived 1).map A.g,
    δ₀ F A,
    F.map A.f, F.map A.g, (0 : F.obj A.3 ⟶ F.obj A.3)]
```

In particular,

$$\begin{tikzcd}
0 \arrow[rr] \arrow[d, dash, "\cong"] & & I \otimes_R M \arrow[r] & R \otimes_R M \\
\mathrm{Tor}_1^R(R/I, M) \arrow[urr, "\delta_0"] & & &
\end{tikzcd}$$

is exact. Since this is true for any finitely generated ideal $I$, (8) implies (4). So finally, all definitions are equivalent.

$$\begin{tikzcd}
(1) \arrow[r, Leftrightarrow] \arrow[ddrr, Rightarrow] & (2) \arrow[r, Leftrightarrow] & (3) \arrow[d, Leftrightarrow] \\
(8) \arrow[u, Rightarrow] & & (4) \\
(7) \arrow[u, Rightarrow] & (6) \arrow[l, Rightarrow] & (5) \arrow[l, Rightarrow]
\end{tikzcd}$$