Regular Paper

可換環上の加群の性質
Properties of Modules over a Commutative Ring

渡瀬 泰成 [1,*]
Yasushige Watase[1,*]

1 東京都杉並区
1 Suginami-ku, Tokyo, Japan

* yasushige.watase@gmail.com

Proof checked by Mizar Version: 8.1.09 and MML Version: 5.57.1355
Received: December 16, 2019. Accepted: April 28, 2020.

Abstract

This article reports current progress of formalizing on theory of modules over a commutative ring. The theory has been formalized in MML appeared as [1–6], however the previous work does not fully cover the theory and still remains many important theorems and notions such as Nakayama's lemma, tensor products and a chain of modules for further formalization. Some common notions such as a faithful module, an annihilator of a module, an exaxt sequence of modules have not been formalized. The aim at this article is to formalize such unformalized subjects. Modules over a ring has a close linkage to the Category Theory and this article tried to deploy some treatment on the language of Category of modules such as the initial/terminal object.

1 はじめに

K-理論のミルナー予想を解決したボエボドスキー (Vladimir Aleksandrovich Vojevodskij, 1966-2017）は, タイプ理論ベースの定理証明支援システム Coq で圏論の言語を用いてホモトピー理論の形式化が可能と考え, 論理の枠組みの見直しを提唱していた. Mizar での定理形式化と圏論の言語の親和性を高めることは出来ないかという着想のもと, 加群の圏の言語を援用して加群の諸性質の形式化を行うことが本稿が目指すところである.

可換環上の加群は [1–6] 等により形式化されて来たが, 加群の諸性質は十分に形式化されてはいない. 加群のテンソル積, 加群の列、完全列は未だ取り扱われてない. 本稿では加群の未だ形式されていない基本的な概念を扱う. 加群とイデアルの積や加群の零化イデ

アルの定義と関連命題, 圏論との整合性のため始対象, 終対象の形式化, さらに加群列を加群の圏の射より完全列, 単完全系列の諸定理の形式化を目指す. 参考文献としては [7] の 2 章, [8] の 1 章の加群, 4 章の圏の内容を参考とした. R 加群全体はもはや集合を為さなず類 (Class) となることが知られる. 直観的には加群全体が集合とすると、加群全体の集合から自由加群を構成すると, 加群全体の集合からはみ出してしまうからと理解される. この点については Mizar では普遍集合 (Universe) [4] を利用して対応している. 本稿に於いてもなるべく一般な形での形式化を目指し加群全体の集合は普遍集合として扱う立場をとる. 加群の圏を扱った先行形式アーティクル [9] の結果も援用する. ただし Mizar で形式化された加群の圏は完全に定義が完了しておらず, 単位法則を欠いた形である. その意味で加群の圏の理論は未整備と言わざる得ないが, 可能な限りこのギャップを埋める工夫を検討する.

2 既存定義類

先行して形式化されてた加群の関連事項について整理し必要な定義類をまとめる. これは加群と加群の圏の Mizar に於ける先行アーティクル [1, 2, 4, 9–11] が 1990 年から 91 年かけ立て続けに形式化され, それらが互いに関連し理解しずらいと判断したためである. 目標としているのは左 R 加群の列の形式化であるのでそれに必要な言語が必要な為である. 加群列は加群の族と加群間の射から構成される. 既に定義されている定義、新規に入用な定義を分けて整理する. 「左 R 加群の圏」を以下 $\mathscr{C}(LMOD_{\mathrm{R}})$ と表記し, 合わせて左 R 加群の圏の対象の集合, 射の集合を夫々 $Ob(LMOD_{\mathrm{R}})$ と $Mor(LMOD_{\mathrm{R}})$ と表記する.

2.1 加群の定義群

加群の定義群で基本的なものを列挙する. 後述の圏での扱いが大いに異なるゆえである.

2.1.1 加群の定義

復習として扱いやすい定義として R 加群は [2] の定義を掲載する.

Listing 1. 左加群の構造-VECT_2-def 5

```
definition
  let R;
  func LeftModule R -> Abelian add-associative right_zeroed
  right_complementable strict non empty ModuleStr over R equals
:: VECTSP_2:def 5
  ModuleStr (#
    the carrier of R, the addF of R, 0.R, the multF of R #);
end;
```

2.1.2 部分加群の生成

環 R のスカラー倍を持つ加群 V の元と R の積を元とし有限個を除きすべてゼロ（加群の零元として）となる列（線形の組み合わせ）を関数により定義し, これを Linear Combination と称する.

Listing 2. 線形組み合わせ (Linear_Combination) VECTSP_6-def 1

```
definition
  let GF be non empty ZeroStr;
  let V be non empty ModuleStr over GF;

  mode Linear_Combination of V ->
       Element of Funcs(the carrier of V, the carrier of GF) means
:: VECTSP_6:def 1

  ex T being finite Subset of V st
  for v being Element of V st not v in T holds it.v = 0.GF;
end;
```

加群の部分集合 A をとり, A の元による Linear Combination L から, 各成分を足し合わせる線形結合 $Sum(L)$ すべてからなる集合により A が生成する部分加群が定義される.

Listing 3. A が生成する部分加群 (Lin(A)) VECT_7-def 2

```
definition
  let GF be Ring;
  let V be LeftMod of GF,
      A be Subset of V;
  func Lin(A) -> strict Subspace of V means
:: VECTSP_7:def 2
  the carrier of it = the set of all Sum(l) where
    l is Linear_Combination of A;
end;
```

2.1.3 加群の準同形

R 加群の準同形は加法群の準同形に以下のスカラー倍の両立の条件を加えたものであり非形式な表現では R 準同形を以下の様に定義する.

定義 1. M, N を R 加群とする. M から N の中への写像 $f : \mathrm{M} \to \mathrm{N}$ が R 準同形であるとは,、任意の $a \in \mathrm{R},\ x, y \in \mathrm{M}$ に対して、

$$f(x + y) = f(x) + f(y), \quad f(ax) = af(x)$$

が成り立つことをいう.

しかしながら、現状の MML ライブラリ-では以下の様に 2 つの定義に分かれている. 加法群の準同形は [12], スカラー倍は [4] で以下の如く定義される.

Listing 4. 加法群の準同形　MOD_4

```
definition let G,H be AddGroup;
  mode Homomorphism of G,H is additive Function of G,H;
end;
```

Listing 5. スカラー倍の保存 (homegeneous) MOD_2-def 2

```
definition
  let R be non empty multMagma;
  let G,H be non empty ModuleStr over R;
  let f be Function of G,H;
  attr f is homogeneous means
:: MOD_2:def 2

  for a being Scalar of R, x being Vector of G holds f.(a*x) = a*f.x;
end;
```

2.2 $\mathscr{C}(LMOD_{\mathrm{R}})$(左 R 加群の圏) の定義群

$\mathscr{C}(LMOD_{\mathrm{R}})$ の構成は, 対象とよばれるものと射により定義される. 対象と射はともに構造体により定義される. それらの上に構造体 CatStr として定義される. $\mathscr{C}(LMOD_{\mathrm{R}})$ から別の圏への関手は本アーティクルでは論じないので, 対象と射の扱いが主な論点となる.

2.2.1 $\mathscr{C}(LMOD_{\mathrm{R}})$ の対象

R 加群の対象の全体を以下の様に可換群とスカラ倍を与える関数のペアとして定義する.

Listing 6. 加群の圏の対象 MODCAT_1-def 6

```
definition
  let UN,R;
  func LModObjects(UN,R) -> set means
:: MODCAT_1:def 6
  for y being object holds y in it iff
    ex x st x in the set of all [G,f] where G is Element of GroupObjects(UN),
    f is Element of Funcs([:the carrier of R,the carrier of G:],
                  the carrier of G)
      & GO x,y,R;
end;
```

2.2.2 $\mathscr{C}(LMOD_{\mathrm{R}})$ の射

R 加群の射は, 始域 (Dom), 終域 (Cod), 始域から終域への関数の組として構造を与え関数が加法とスカラ倍を保つもの (LModMorphism-like) として定義する.

Listing 7. 加群の圏の射の構造 MOD_2 Morphism Str

```
definition
  let R;
  struct LModMorphismStr over R (# Dom,Cod -> LeftMod of R, Fun -> Function of
    the Dom, the Cod #);
end;

definition
  let R;
  let IT be LModMorphismStr over R;
  attr IT is LModMorphism-like means
:: MOD_2:def 7

  fun(IT) is additive homogeneous;
end;
```

2.2.3 $\mathscr{C}(LMOD_{\mathrm{R}})$ の構造体

ここでは、圏の構造体の定義を述べる．圏の公理では任意の射と合成しても射を変えない単位射を単位法則といい (Mizar では with_identities と記述), 単位射の存在を仮定する．加群の圏の形式化 [9] では単位法則は形式化されていない．

Listing 8. 加群の圏の構造 MODCAT_1-ModCatStr

```
definition
  let UN,R;
  func LModCat(UN,R) −> strict CatStr equals
  CatStr(#LModObjects(UN,R),Morphs(
    LModObjects(UN,R)), dom(LModObjects(UN,R)),cod(LModObjects(UN,R)), comp(
    LModObjects(UN,R))
   #);
  coherence;
end;
```

2.2.4 $\mathscr{C}(LMOD_{\mathrm{R}})$ の始対象、終対象

圏論の一般論として始対象や終対象の概念がある．加群の圏は実際、始対象, 終対象をもつ事の形式化を目標にする．定義としては夫々以下の通りである．

定義 2. 対象 $I \in Ob(\mathscr{C})$ が始対象 *initial object* であるとは, 任意の対象 $A \in Ob(\mathscr{C})$ に対して

$$\exists f_A \in Mor(\mathscr{C}) \ s.t \ Hom(I, A) = \{f_A\}$$

定義 3. 対象 $T \in Ob(LMOD_{\mathrm{R}})$ が終対象 *terminal object* であるとは, 任意の対象 $A \in Ob(LMOD_{\mathrm{R}})$ に対して

$$\exists g_A \in Mor(LMOD_{\mathrm{R}}) \ s.t \ Hom(A, T) = \{g_A\}$$

と定義される．これに対応する形式化は以下の定義となる．定義 2, 3 と形式化の定義は異なるが, 同値な言い換えである．

Listing 9. 圏の終対象と始対象 CAT_1-def18 19

```
definition
  let C,a;
  attr a is terminal means
:: CAT_1:def 18

  Hom(b,a)<>{} & ex f being Morphism of b,a
  st for g being Morphism of b,a holds f=g;
  attr a is initial means
:: CAT_1:def 19

  Hom(a,b)<>{} & ex f being Morphism of a,b st
  for g being Morphism of a,b holds f=g;
```

2.3 新規に形式化する定義や用語

加群理論として有用な定義として以下が挙げられる．

- R のイデアルと R 加群の積,
- R 加群 M, N の割り算の類似物 M : N と M の零化イデアル,
- 左加群としての M と N の和と交わり M + N と M ∩ N,
- $\mathscr{C}(LMOD_{\mathrm{R}})$ の始対象, 終対象,
- R 加群の列.

2.3.1 R のイデアルと R 加群の積

$\mathfrak{a}$ を R のイデアルとし、M を R 加群としたとき、$\mathfrak{a}$ と M の積を以下のように定義する. 加群の線形組み合わせの関数 ($Linear\_Comibnation$) を利用すると成分の和をとる関数 (Sum) の変更点が多くなってしまう.

定義 4.

$$\mathfrak{a} \cdot \mathrm{M} = \left\{ \sum_{finite} a_i \cdot m_i \mid a_i \in \mathfrak{a},\ m_i \in \mathrm{M}) \right\}$$

関数 Sum の変更を避け形式化では以下の様に定義した.

Listing 10. イデアルと加群の積 LMODSEQ1-def1

```
definition
  let R,I,J;
  func Mult_(I,J)−> Subset of J equals
:: LMODSEQ1:def 1
  { Sum s where s is FinSequence of the
  carrier of J : for i being Element of NAT st 1 <= i & i <= len s
  ex a be Element of I, b being Element of J st s.i = a∗b };
end;
```

2.3.2 R 加群 M, N の割り算の類似物 M : N と M の零化イデアル

R 加群 M, N から R のイデアルを以下の様に定義し M : N と表記する.

定義 5.

$$\mathrm{M} : \mathrm{N} = \{a \in \mathrm{R} \mid a \cdot n \ \in \mathrm{M}\ where\ n \in \mathrm{N}\ \}$$

特に R 加群 N が零加群のとき M : N を M の零化イデアルといい $Ann(\mathrm{M})$ と表記する.

Listing 11. 加群の割り算の類似物 (N:M) と零化イデアル (Ann(M)) LMODSEQ1:def 2–3

```
definition
  let R,M,N,P;
  func N[:]P −> Subset of the carrier of R equals
:: LMODSEQ1:def 2
  { a where a is Element of R: for p being Element of P holds a∗p in N };
end;

definition
  let R,M;
  func Ann(M) −> Subset of the carrier of R equals
```

```
:: LMODSEQ1:def 3
  { a where a is Element of R: for p being Element of M holds a*p in (0).M };
end;
```

2.3.3 左加群として $M + N$ と $M \cap N$

既存アーティクル [13] が加群の和や交わりを部分空間として扱っており基本的な形式化である．しかし内容はアーティクル RMOD_3 を基に構築されており直接左加群としては扱えない．部分空間として出来上がった対象を左加群と同一視する．FM 誌にはアーティクル [6] があり，その FM 誌での識別子 (Identifier) は LMOD_3 である，しかしながら実際の MML ライブラリには RMOD_3 しかなく混乱の原因である．本編では左加群からの部分加群の和と交わりの構成を形式化し，左加群だけで理論が構成出来るように一貫性を持たせた．

Listing 12. 部分加群の和と交わり LMODSEQ1:def 5-6

```
definition
  let R, M,W1,W2;
  func W1 + W2 -> strict Subspace of M means
:: LMODSEQ1:def 5

  the carrier of it = {v + u : v in W1 & u in W2};
end;

definition
  let R, M, W1,W2;
  func W1 /\ W2 -> strict Subspace of M means
:: LMODSEQ1:def 6

  the carrier of it = (the carrier of W1) /\ (the carrier of W2);
  commutativity;
end;
```

以上の定義から M : N と零化イデアルに関連する幾らかの命題が以下の様に形式化される．

Listing 13. 命題群 LMODSEQ1:4–6

```
theorem :: LMODSEQ1:4
   for R,M,N,P st N = (0).M & P = M holds Ann(M) is Ideal of R;

theorem :: LMODSEQ1:5
   for R,M,N,W st N = (0).W holds Ann(W) = N[:]W;


theorem :: LMODSEQ1:6
   for R,M,N,W st N = (0).M holds Ann(W) = N[:]W;
```

2.3.4 $\mathscr{C}(LMOD_{\mathrm{R}})$ の始対象、終対象

加群の圏での始対象，終対象形式化を行うが，定義 2, 3 に準じて定義する．圏の射の定義による定義では具体的な準同形を経由した定義とした．圏のレベルから集合のレベルへの橋渡しの役目を負わせた以下の形で形式化する．

Listing 14. 加群圏の終対象と始対象 LMODSEQ1:def 8-def 9

```
definition
  let R,a;
  attr a is terminal means
:: LMODSEQ1:def 8

  Hom(b,a)<>{} & ex f being Morphism of b,a
  st for g being Function of b,a st g is additive & g is homogeneous
  holds fun(f) = g;
  attr a is initial means
:: LMODSEQ1:def 9

  Hom(a,b)<>{} & ex f being Morphism of a,b st
  for g being Function of a,b st g is additive & g is homogeneous
  holds fun(f) = g;
end;
```

2.3.5　加群の列

ここでは加群の列を具体的に定義する. 非形式な加群の列の定義は Λ を添え字の集合として $\{M_\lambda\}\ \lambda \in \Lambda$ と表現する. 先ずは添え字の集合を自然数の集合に取り加群の列を考察して行く. 以下様な加群の列 $\{\mathrm{M}_i\}$ と M_i と M_{i+1} 間の準同形写像 f_i によって決まる.

$$\mathrm{M}_1 \xrightarrow{f_1} \mathrm{M}_2 \xrightarrow{f_2} \cdots \xrightarrow{f_{i-1}} \mathrm{M}_i \xrightarrow{f_i} \mathrm{M}_{i+1} \xrightarrow{f_{i+1}} \cdots \tag{1}$$

前章で見た加群の圏で定義された射により図式 (1) の表現を工夫する. 射がきまると始域 *dom* と終域 *cdom* が指定されるので $\{\mathrm{M}_i\}$ を表記せず以下の様に表現する.

$$\cdot \xrightarrow{f_1} \cdot \xrightarrow{f_2} \cdots \xrightarrow{f_{i-1}} \cdot \xrightarrow{f_i} \cdot \xrightarrow{f_{i+1}} \cdots \tag{2}$$

この図式 (2) を単純な射の列の構成とすると、

Listing 15. 加群の圏の射の列 LMODSEQ1-Module Sequence

```
definition
  let UN,A;
  mode Mor_Seq of UN,A is sequence of Morphs(LModObjects(UN,A));
end;
```

矢印間に現れる点、$\rightarrow \cdot$(この点) $\rightarrow$ は本来 M_i となるべきところだが射 f_{i+1} の始域は f_i の終域に一致するので、$\mathrm{cod}\ f_i = \mathrm{dom}\ f_{i+1}$ なる条件が課して加群の列とする.

Listing 16. 加群の列 LMODSEQ1-def 10

```
definition
  let R,UN;
  let F be Mor_Seq of UN,R;
  attr F is Module_Sequence means
:: LMODSEQ1:def 10
  for i be Nat holds cod'(F.i) = dom'(F.(i+1));
end;
```

2.4　作成予定の形式化すべき事項

『R 加群』の形式化を試みたが、その途に就いたばかりで困難点も顕在化した. 圏論の用語を援用しながらの形式化を試みようとしたが、加群の圏の定義も不完全のままであ

り、左右の加群で定義が乖離してしまう事実もすでに述べた．しかしながらこれらの原因と是正を試みることも重要であり本編の意義もあろう．始対象、終対象の形式化には、ともかくも射の構造に埋め込まれている準同形写像の性質を表現する工夫が必要である．このために準同形写像から出発して始対象なり, 終対象なりの構成を行いたい.

Listing 17. 命題 MODSEQ1 - Th14

```
theorem :: LMODSEQ1:14
  g is additive & g is homogeneous implies g = ZeroMap(TrivialLMod(R),b);
```

このための定理が Th15 となる．ここで現れる準同形写像 ZeroMap をもとに始対象, 終対象を構成する．ここで具体的な準同形写像の内容を $\mathscr{C}(LMOD_{\mathrm{R}})$ の射に取り込む方法を確立できればよいのである．以下の完全列では準同形の内容が肝となる．

2.4.1 加群の完全列

ここでは加群の列に完全の条件を付加する過程を考察する．

定義 6. 図式 (1) が完全であるとは, $\forall i \in \mathbb{N},\ \mathrm{Im} f_i = \mathrm{Ker} f_{i+1}$ をみたすことをいう.

記述として単純な定義となるが、実際に List:MODSEQ1:def10 で指定される加群の射 $F(i)$ は, そのまま準同形を表しておらず $F(i)$ は射の構造体であり, 任意の準同形 f が $rngf \subseteq codF_i$ をみたしていることしか情報をもたない．よって射の構造に何らかの条件をつける工夫が必要となる．例えば Onto-Morphism とか Into-Morphism など Attribute を加える方策も検討する．

3 考察と展望

前章では困難な点を課題として述べた．圏とは関係なしに重要な加群の定理, 中山の補題などの証明は形式化可能である．他方テンソル積や Hom 加群は圏論の関係が深く関手としての役割を担っている．

3.1 有限生成加群の定理として中山の補題

定理 1. 中山の補題 M を有限生成加群、$\mathfrak{a}$ はジャコブソン根基に含まれるとき以下のが成立する．

$$\mathfrak{a} \cdot \mathrm{M} = 0$$

この証明には行列で習うケーリー・ハミルトンの定理の加群版が入用となり道のりは長いが, ジャコブソン根基も既に [14] にて準備してあるので十分形式化可能な目標である．また代数的数の為す集合が環となることも導出されるので, 代数的整数の理論の [15] の続編が可能となる．

3.2 テンソル積 $\otimes$

加群同士の和は 2 章 2 節にて論じた様に既に形式化されている．しかし加群同士の積は加群とはならにので, 直積集合とって加群になるような同値関係で割って構成されるのがテンソル積である．これを $\mathscr{C}(LMOD_{\mathrm{R}})$ の双対積という．テンソル積は加群の係数環を拡大するための関手でもあり, 形式化のあとには圏論的取り扱いを想定して形式化の構想, 設計時点で圏論の要素を取り入れる．

3.3 Hom 加群

任意の R 加群 M, N に対して M から N への準同形全体を $Hom(\mathrm{M}, \mathrm{N})$ と表記するが, これが R 加群になることを形式化する．これは加群の特徴でもあるので重要である．また加群の列に作用する関数などに用いられるので上手に関数として定義する必要がある．

3.4 $\mathscr{C}(LMOD_{\mathrm{R}})$ に単位法則を導入

$\mathscr{C}(LMOD_{\mathrm{R}})$ は [9] により $CatStr$ が導入され、$CatStr$ が $Categorylike$ であることを形式化し, それらが transitive, associative, reflexive の 3 性質をみたすことを証明して終わる．何故「単位法則 (with_identities)」が導入できないかを解明し $\mathscr{C}(LMOD_{\mathrm{R}})$ が圏なることを完成させる．

3.5 展望

圏が直積, 双対積をもつこと, 始対象と終対象が一致すること, 核, 余核が存在 (定義は省く) 存在すること, 準同形定理が成り立つことを満たすときアーベル圏という．$\mathscr{C}(LMOD_{\mathrm{R}})$ はアーベル圏の一例となる．層の理論はアーベル圏の理論の応用とみなされるので, 位相空間上の加群の層が展開で出来るようになること期待している．圏の言語を取り入れて記述する形式化の工夫やその数理論理的根拠を今後考慮に入れる必要があると考える．

参考文献

[1] Kusak E, Leończuk W, Muzalewski M. Abelian Groups, Fields and Vector Spaces. Formalized Mathematics. 1990;1(**2**):335–342. Available from: `http://fm.mizar.org/1990-1/pdf1-2/vectsp_1.pdf`.

[2] Muzalewski M. Construction of Rings and Left-, Right-, and Bi-Modules over a Ring. Formalized Mathematics. 1991;2(**1**):3–11. Available from: `http://fm.mizar.org/1991-2/pdf2-1/vectsp_2.pdf`.

[3] Trybulec WA. Linear Combinations in Vector Space. Formalized Mathematics. 1990;1(**5**):877–882. Available from: `http://fm.mizar.org/1990-1/pdf1-5/vectsp_6.pdf`.

[4] Muzalewski M. Rings and Modules – Part II. Formalized Mathematics. 1991;2(**4**):579–585. Available from: `http://fm.mizar.org/1991-2/pdf2-4/mod_2.pdf`.

[5] Muzalewski M. Submodules. Formalized Mathematics. 1992;3(**1**):47–51. Available from: `http://fm.mizar.org/1992-3/pdf3-1/lmod_6.pdf`.

[6] Muzalewski M, Skaba W. Operations on Submodules in Left Module over Associative Ring. Formalized Mathematics. 1991;2(**2**):289–293. Available from: `http://fm.mizar.org/1991-2/pdf2-2/lmod_3.pdf`.

[7] Atiyah MF, Macdonald IG. Introduction to Commutative Algebra. vol. 2. Addison-Wesley Reading; 1969.

[8] 河田 敬義: ホモロジー代数. 岩波書店; 1977.

[9] Muzalewski M. Category of Left Modules. Formalized Mathematics. 1991;2(**5**):649–652. Available from: `http://fm.mizar.org/1991-2/pdf2-5/modcat_1.pdf`.

[10] Muzalewski M, Skaba W. Submodules and Cosets of Submodules in Left Module over Associative Ring. Formalized Mathematics. 1991;2(**2**):283–287. Available from: `http://fm.mizar.org/1991-2/pdf2-2/lmod_2.pdf`.

[11] Muzalewski M, Skaba W. Linear Combinations in Left Module over Associative Ring. Formalized Mathematics. 1991;2(**2**):295–300. Available from: `http://fm.mizar.org/1991-2/pdf2-2/lmod_4.pdf`.

[12] Muzalewski M. Opposite Rings, Modules and their Morphisms. Formalized Mathematics. 1992;3(**1**):57–65. Available from: `http://fm.mizar.org/1992-3/pdf3-1/mod_4.pdf`.

[13] Trybulec WA. Operations on Subspaces in Vector Space. Formalized Mathematics. 1990;1(**5**):871–876. Available from: `http://fm.mizar.org/1990-1/pdf1-5/vectsp_5.pdf`.

[14] Watase Y. Zariski Topology. Formalized Mathematics. 2018;26(**4**):277–283.

[15] Watase Y. Algebraic Numbers. Formalized Mathematics. 2016;24(**4**):291–299.

Mizar article information

Works in Progress

LMODSEQ1 Properties of Modules over a Commutative Ring
by Yasushige Watase
Summary: This article discusses current progress of formalizing on theory of modules over a commutative ring. The theory has been formalized in MML appeared as [1–6], however the previous work does not fully cover the theory and still remains many

important theorems and notions such as Nakayama's lemma, a chain of modules for further formalization. Some common notions such as a faithful module, an annihilator of a module, a sequence of modules have not been formalized. The aim at this article is to formalize such unformalized theorems. Modules over a ring has a close linkage to Category Theory and this article deploys some treatment on the language of Category of modules such as the initial/terminal object.

Listing 18. LMODSEQ1 - abstract

```
:: Sequence of R-Module
environ
 vocabularies TARSKI, FUNCSDOM, VECTSP_1, STRUCT_0, XBOOLE_0, VECTSP_2,
     NAT_1, SUBSET_1, FUNCT_1, RELAT_1, FUNCT_2, CLASSES2, MOD_2, MODCAT_1,
     FUNCOP_1, UNIALG_1, CAT_1, GRCAT_1, ENS_1, NUMBERS, ARYTM_3, GROUP_6,
     MSSUBFAM, CARD_3, FINSEQ_1, XXREAL_0, CARD_FIL, RMOD_2, SUPINF_2,
     RSSPACE, RLSUB_1, IDEAL_1,
     LMODSEQ1;

 notations TARSKI, XBOOLE_0, SUBSET_1, RELAT_1, FUNCT_1, FUNCT_2, FUNCOP_1,
     NUMBERS, XXREAL_0, NAT_1, CLASSES2, FINSEQ_1, STRUCT_0, ALGSTR_0,
     RLVECT_1, VECTSP_1, VECTSP_2, VECTSP_4, GRCAT_1, MOD_2, MODCAT_1, IDEAL_1;

 constructors REALSET1, FUNCT_5, RELSET_1, RING_2, GRCAT_1, MODCAT_1;

 registrations ORDINAL1, XREAL_0, STRUCT_0, RELSET_1, SUBSET_1, GRCAT_1,
     MOD_2, IDEAL_1, RLVECT_1, VECTSP_4;

 requirements REAL, SUBSET, BOOLE, ARITHM, NUMERALS;

 definitions XBOOLE_0, TARSKI;

 equalities XBOOLE_0, ALGSTR_0, MOD_2, STRUCT_0;

 expansions TARSKI, STRUCT_0, VECTSP_1, IDEAL_1;

 theorems TARSKI, FUNCT_2, FINSEQ_1, VECTSP_1, XBOOLE_0,
     NAT_1, XREAL_1, VECTSP_4, RELAT_1, GRCAT_1, FUNCT_1, FUNCOP_1, RLVECT_1,
     IDEAL_1, XBOOLE_1;

begin
reserve UN for Universe;
reserve R for non degenerated comRing;
reserve f for LModMorphismStr over R;
reserve o,o1,x,y, x,y,y1,y2 for object;
reserve a,b, M,J,V,G,G1,H,H1,T for LeftMod of R;
reserve q,I for Ideal of R;
reserve u,u1, u2,v,v1,v2 for Vector of M;

definition let G,H be AddGroup;
  mode Homomorphism of G,H is additive Function of G,H;
end;

definition
  let R be Ring;
  let J be LeftMod of R;
  mode Submodule of J is Subspace of J;
end;

reserve N,P,W,W1,W2,W3 for Submodule of M;

::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::: a*I [AM] page 19
::::::::::::::::::::::::::::::::::::::::::::::::::::::::

definition
  let R,I,J;
```

```
  func Mult_(I,J)−> Subset of J equals
:: LMODSEQ1:def 1
  { Sum s where s is FinSequence of the
  carrier of J : for i being Element of NAT st 1 <= i & i <= len s
  ex a be Element of I, b being Element of J st s.i = a∗b };
end;

registration
  let R,I,J;
  cluster Mult_(I,J) −> non empty;
end;

::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::: To be proven Mult_(I,J) is Module
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::: N:P [AM] page 19
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

definition
  let R,M,N,P;
  func N[:]P −> Subset of the carrier of R equals
:: LMODSEQ1:def 2
  { a where a is Element of R: for p being Element of P holds a∗p in N };
end;

registration
  let R,M,N,P;
  cluster N[:]P −> non empty;
end;

theorem :: LMODSEQ1:1
    for R,M,N,P holds
    N[:]P is Ideal of R;

::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
::: Annihilator of M [AM] page 20 Exercise 2.2
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

definition
  let R,M;
  func Ann(M) −> Subset of the carrier of R equals
:: LMODSEQ1:def 3
  { a where a is Element of R: for p being Element of M holds a∗p in (0).M };
end;

registration
  let R,M;
  cluster Ann(M) −> non empty;
end;

theorem :: LMODSEQ1:2
    for R,M,N,P st N = (0).M & P = M holds N[:]P is Ideal of R;

theorem :: LMODSEQ1:3
    for R,M,N,P st N = (0).M & P = M holds Ann(M) = N[:]P;

theorem :: LMODSEQ1:4
    for R,M,N,P st N = (0).M & P = M holds Ann(M) is Ideal of R;

theorem :: LMODSEQ1:5
    for R,M,N,W st N = (0).W holds Ann(W) = N[:]W;


theorem :: LMODSEQ1:6
    for R,M,N,W st N = (0).M holds Ann(W) = N[:]W;

definition
```

```
  let R,M;
  attr M is faithfull means
:: LMODSEQ1:def 4
  Ann(M) = {0.R} ;
end;

::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
:::: rewrite form right-module W1+W2 & W1 /\ W2 case to those of left-module
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
definition
  let R, M,W1,W2;
  func W1 + W2 -> strict Subspace of M means
:: LMODSEQ1:def 5

  the carrier of it = {v + u : v in W1 & u in W2};
end;

definition
  let R, M, W1,W2;
  func W1 /\ W2 -> strict Subspace of M means
:: LMODSEQ1:def 6

  the carrier of it = (the carrier of W1) /\ (the carrier of W2);
  commutativity;
end;

theorem :: LMODSEQ1:7
    for R, M, W1,W2 holds o in W1 implies o in W1 +W2;

theorem :: LMODSEQ1:8
    for R, M, W1,W2 holds o in W2 implies o in W1 +W2;

theorem :: LMODSEQ1:9
   for R, M, W1,W2 holds W1 is Subspace of W1 + W2 & W2 is Subspace of W1 + W2;

theorem :: LMODSEQ1:10
    for R,M,N,W1,W2 st N = (0).M
    holds Ann(W1+W2) = N[:](W1+W2);

theorem :: LMODSEQ1:11
    for R,M,N,W1,W2 st N = (0).M
    holds Ann(W1 /\ W2) = N[:](W1 /\ W2);

theorem :: LMODSEQ1:12
  for R,M,N,W1,W2 holds N[:](W1+W2) = N[:]W1 /\ N[:]W2;

theorem :: LMODSEQ1:13
   for R,M,W1,W2 holds
   Ann(W1+W2) = Ann(W1) /\ Ann(W2);

::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
:::: Sequence of R-Module
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

definition
::registration
  let R,G,H;
  func Hom(G,H) -> non empty Subset of Maps(G,H) equals
:: LMODSEQ1:def 7
  {f where f is Element of Maps(G,H) : f is homogeneous};
end;

:::Nedd to copy the ZERO(G,H) from MOD_2
definition
  let R,G,H;
  redefine func ZERO(G,H) -> strict Morphism of G,H;
end;

reserve g for Function of TrivialLMod(R),b;
```

```
theorem :: LMODSEQ1:14
  g is additive & g is homogeneous implies g = ZeroMap(TrivialLMod(R),b);

definition
  let R,a;
  attr a is terminal means
:: LMODSEQ1:def 8


  Hom(b,a)<>{} & ex f being Morphism of b,a
  st for g being Function of b,a st g is additive & g is homogeneous
  holds fun(f) = g;
  attr a is initial means
:: LMODSEQ1:def 9


  Hom(a,b)<>{} & ex f being Morphism of a,b st
  for g being Function of a,b st g is additive & g is homogeneous
  holds fun(f) = g;
end;

:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
:::: To be proven TrivialLMod(R) is initial & terminal Object
:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
:::: Sequence of R−Module
:::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
reserve i,j,k for Element of NAT;

definition
  let UN,R;
  mode Mor_Seq of UN,R is sequence of Morphs(LModObjects(UN,R));
end;

reserve F for Mor_Seq of UN,R;

definition
  let R,UN;
  let F be Mor_Seq of UN,R;
  attr F is Module_Sequence means
:: LMODSEQ1:def 10
  for i be Nat holds cod'(F.i) = dom'(F.(i+1));
end;
```