Formalising the Proj Construction in Lean

Jujian Zhang
Department of Mathematics, Imperial College London, UK

Abstract

Many objects of interest in mathematics can be studied both analytically and algebraically, while at the same time, it is known that analytic geometry and algebraic geometry generally do not behave the same. However, the famous GAGA theorem asserts that for projective varieties, analytic and algebraic geometries are closely related; the proof of Fermat's last theorem, for example, uses this technique to transport between the two worlds [13]. A crucial step of proving GAGA is to calculate cohomology of projective space [12, 8], thus I formalise the Proj construction in the Lean theorem prover for any $\mathbb{N}$-graded R-algebra A and construct projective n-space as $\mathrm{Proj}\, A[X_0, \ldots, X_n]$. This is the first family of non-affine schemes formalised in any theorem prover.

2012 ACM Subject Classification Theory of computation → Logic and verification; Mathematics of computing → Topology

Keywords and phrases Lean, formalisation, algebraic geometry, scheme, Proj construction, projective geometry

Digital Object Identifier 10.4230/LIPIcs.ITP.2023.35

Supplementary Material *Software (Source Code)*: `https://github.com/leanprover-community/mathlib/pull/18138/commits/00c4b0918a2c7a8b62291581b0e1eddf2357b5be`
archived at `swh:1:dir:0876b1af377d6c78a3d76073c0bbd7fe0176d9c6`

Funding *Jujian Zhang*: Schrödinger Scholarship Scheme.

Acknowledgements I want to thank Eric Wieser for his contribution and suggestion in formalising homogeneous ideals and homogeneous localisation; Andrew Yang and Junyan Xu for their review and comments on my code; Kevin Buzzard for suggesting this project and all the contributors to mathlib for otherwise this project would not have been possible.

1 Introduction

Algebraic geometry concerns polynomials and analytic geometry concerns holomorphic functions. Though all polynomials are holomorphic, the converse is not true; thus many analytic objects are not algebraic, for example, $\{x \in \mathbb{C} \mid \sin(x) = 0\}$ can not be defined as the zero locus of a polynomial in one variable, for polynomials always have only finite number of zeros. However, for projective varieties over $\mathbb{C}$, the categories of algebraic and analytic coherent sheaves are equivalent; a consequence of this statement is that all closed analytic subspace of projective n-space $\mathbb{P}_n$ is also algebraic [13, 4]. A crucial step in proving the above statement is to consider the cohomology of projective n-space $\mathbb{P}_n$ [12].

While one can define $\mathbb{P}_n$ over $\mathbb{C}$ without consideration of other projective varieties, it would be more fruitful to formalise the Proj construction as a **scheme** and recover $\mathbb{P}_n$ as $\mathrm{Proj}\, \mathbb{C}[X_0, \ldots, X_n]$, since, among other reasons, by considering different base rings, one may obtain different projective varieties, for example, for any homogeneous polynomials $f_1, \ldots, f_k$, $\mathrm{Proj}\left(\frac{\mathbb{C}[X_0,\ldots,X_n]}{(f_1,\ldots,f_k)}\right)$ defines a projective variety over $\mathbb{C}$.

In this paper I describe a formal construction of $\mathrm{Proj}\, A$ in the Lean3 theorem prover [7] by closely following [9, Chapter II]. The formal construction uses various results from the Lean mathematical library mathlib, most notably the graded algebra and Spec construction; this project has been partly accepted into mathlib already while the remaining part is still

14th International Conference on Interactive Theorem Proving (ITP 2023).
Editors: Adam Naumowicz and René Thiemann; Article No. 35; pp. 35:1–35:17
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

undergoing a review process. The code discussed in this paper can be found on GitHub[1]. I have freely used the axiom of choice and the law of excluded middle throughout this project since the rest of mathlib freely uses classical reasoning as well; consequently, the final construction is not computable. This will not matter for the applications in mind, for example calculating sheaf cohomology and the GAGA theorem.

As previously mentioned, Proj construction heavily depends on graded algebras and the Spec construction. A detailed description of graded algebra in Lean and mathlib, as well as a comparison of graded algebras with that in other theorem provers, can be found in [17]; for my purpose, I have chosen to use an internal grading for any graded ring $A \cong \bigoplus \mathcal{A}_i$ so that the result of the construction is about homogeneous prime ideals of A directly instead of $\bigoplus_i \mathcal{A}_i$. The earliest complete Spec construction in Lean can be found in [2] where the construction followed a "sheaf-on-a-basis" approach from [14, Section 01HR], however, it differs significantly from the Spec construction currently found in mathlib where the construction follows [9, Chapter II]; for this reason, I have also chosen to follow the definition in [9, Chapter II]. Some other theorem provers also have or partially have the Spec construction: in Isabelle/HOL, Spec is formalised by using locales and rewriting topology and ring theory part of the existing library in [1], however, the category of schemes is yet to be formalized; an early formalisation of Spec in Coq can be found in [3] and a definition of schemes in general can be found in its `UniMath` library [16]; due to homotopy type theory of `Agda`, only a partial formalisation of Spec construction can be found in [11]. Though some theorem provers have defined a general scheme, I could not find any concrete construction of a scheme other than Spec of a ring[2]. Thus this paper exhibits the first concrete formalised example of non-affine scheme.

After explaining the mathematical details involved in the Proj construction in Section 2, Lean code will be provided and explained in Section 3. For typographical reasons, some code of formalisation will be omitted and marked as `omitted` or `_` and some code presented in this paper is presented with shortened notations for presentability and readability.

2 Mathematical details

In this section, certain familiarity with basic ring theory, topology and category theory will be assumed. In Sections 2.1 and 2.2, definition of a scheme is explained in detail; Spec construction will also be briefly explained to fix the mathematical approach used in mathlib. Then by following the definition of a scheme step by step, the Proj construction will be explained in Section 2.3.

2.1 Sheaves and Locally Ringed Spaces

Let X be a topological space and $\mathfrak{Opens}(X)$ be the category of open subsets of X.

▶ **Definition 1** (Presheaves [10]). *Let C be a category. A C-*valued presheaf $\mathcal{F}$ on X *is a functor $\mathfrak{Opens}(X)^{\mathsf{op}} \Longrightarrow C$. Morphisms between C-valued presheaves $\mathcal{F}, \mathcal{G}$ are natural transformations. The category thus formed is denoted as $\mathfrak{PSh}(X, C)$.*

In this paper, the category of interest is the category of presheaves of rings $\mathfrak{Psh}(X, \mathfrak{Ring})$. More explicitly, a presheaf of rings $\mathcal{F}$ assigns to each open subset $U \subseteq X$ a ring $\mathcal{F}(U)$ whose elements are called sections on U and for any open subsets $U \subseteq V \subseteq X$, $\mathcal{F}$ assigns

[1] url: `https://github.com/leanprover-community/mathlib/pull/18138/`
[2] In this paper, all rings are assumed to be unital and commutative.

a ring homomorphism $\mathcal{F}(V) \to \mathcal{F}(U)$ often denoted as res_U^V or simply with a vertical bar $s\,|_U$ (a section s on V restricted to U). Examples of presheaves of rings are abundant: considering open subsets of $\mathbb{C}$, $U \mapsto \{$(continuous, holomorphic) functions on $U\}$ with the natural restriction map defines a presheaf of rings. In these examples, compatible sections on different open subsets can be glued together to form bigger sections on the union of the said open subsets; this property can be generalized to arbitrary categories:

▶ **Definition 2** (Sheaves [10, 14]). *A presheaf $\mathcal{F} \in \mathfrak{Psh}(X, C)$ is said to be a* sheaf *if for any open covering of an open set $U = \bigcup_i U_i \subseteq X$, the following diagram is an equalizer*

$$\mathcal{F}(U) \xrightarrow{\left(\mathrm{res}_{U_i}^{U}\right)} \prod_i \mathcal{F}(U_i) \underset{\left(\mathrm{res}_{U_i\cap U_j}^{U_j}\right)}{\overset{\left(\mathrm{res}_{U_i\cap U_j}^{U_i}\right)}{\rightrightarrows}} \prod_{i,j} F(U_i \cap U_j).$$

The category of sheaves *$\mathfrak{Sh}(X, C)$ is the full subcategory of the category of presheaves satisfying the sheaf condition.*

▶ **Definition 3** (Locally Ringed Space [14, 9]). *If $\mathcal{O}_X$ is a sheaf of rings on X, then the pair $(X, \mathcal{O}_X)$ is called a* ringed space*; a* morphism *between two ringed space $(X, \mathcal{O}_X)$ and $(Y, \mathcal{O}_Y)$ is a pair (f, ϕ) such that $f : X \to Y$ is continuous and $\phi : \mathcal{O}_Y \to f_*\mathcal{O}_X$ is a morphism of sheaves where $f_*\mathcal{O}_X \in \mathfrak{Sh}(Y)$ assigns $V \subseteq Y$ to $\mathcal{O}_X(f^{-1}(V))$. A* locally ringed space *$(X, \mathcal{O}_X)$ is a ringed space such that for any $x \in X$, its stalk $\mathcal{O}_{X,x}$ is a local ring where $\mathcal{O}_{X,x} = \mathrm{colim}_{x\in U\in\mathfrak{Opens}X}\, \mathcal{O}_X(U)$; a* morphism *between two locally ringed spaces $(X, \mathcal{O}_X)$ and $(Y, \mathcal{O}_Y)$ is a morphism (f, ϕ) of ringed space such that for any $x \in X$ the ring homomorphism induced on stalk $\phi_x : \mathcal{O}_{Y,f(x)} \to \mathcal{O}_{X,x}$ is local.*

From the previous definitions, if $\mathcal{O}_X$ is a presheaf and $U \subseteq X$ is an open subset, then there is a presheaf $\mathcal{O}_X\,|_U$ on U by assigning every open subset V of U to $\mathcal{O}_X(V)$. This is called restricting a presheaf; sheaves, ringed spaces and locally ringed spaces can also be similarly restricted.

2.2 Definition of Affine Scheme and Scheme

The Spec construction

Let R be a ring and let $\operatorname{Spec} R$ denote the set of prime ideals of R. Then for any subset $s \subseteq R$, its zero locus is defined as $\{\mathfrak{p} \mid s \subseteq \mathfrak{p}\}$. These zero loci can be considered as closed subsets of $\operatorname{Spec} R$; the topology thus formed is called the Zariski topology. Then a sheaf of rings on $\operatorname{Spec} R$ can be defined by assigning $U \subseteq \operatorname{Spec} R$ to the ring

$$\left\{ s : \prod_{x\in U} R_x \mid s \text{ is locally a fraction} \right\},$$

where s is *locally a fraction* if and only if for any prime ideal $x \in U$, there is always an open subset $x \in V \subseteq U$ and $a, b \in R$ such that for any prime ideal $y \in V$, $b \notin y$ and $s(y) = \frac{a}{b}$. This sheaf $\mathcal{O}$ is called the *structure sheaf* of $\operatorname{Spec} R$. $(\operatorname{Spec} R, \mathcal{O})$ is a locally ringed space because for any prime ideal $x \subseteq R$, $\mathcal{O}_x \cong A_x$ [9, Chapter 2, Proposition 2.2].

Scheme

▶ **Definition 4** (Scheme). *A locally ringed space $(X, \mathcal{O}_X)$ is said to be a* scheme *if for any $x \in X$, there is always some ring R and some open subset $x \in U \subseteq X$ such that $(U, \mathcal{O}_X\,|_U) \cong (\operatorname{Spec} R, \mathcal{O}_{\operatorname{Spec} R})$ as locally ringed spaces. The* category of schemes *is the full subcategory of locally ringed spaces where objects are schemes.*

Thus to construct a scheme, one needs the following:

- a topological space X;
- a presheaf $\mathcal{O}$;
- a proof that $\mathcal{O}$ satisfies the sheaf condition;
- a proof that all stalks are local;
- an open covering $\{U_i\}$ of X;
- a collection of rings $\{R_i\}$ and isomorphism $(U_i, \mathcal{O}_X|_{U_i}) \cong (\operatorname{Spec} R_i, \mathcal{O}_{\operatorname{Spec} R})$.

In Section 2.3, the Proj construction will be described following the steps above. Hence, the Proj construction though appears to be a definition, is in fact a mixture of defining a ringed space and a proof that the constructed ringed space is locally affine.

2.3 The Proj Construction

Throughout this section, R will denote a ring and A an $\mathbb{N}$-graded R-algebra, in order to keep notations the same as Section 3, the grading of A will be written as $\mathcal{A}$, i.e. $A \cong \bigoplus_{i\in\mathbb{N}} \mathcal{A}_i$ as R-algebras.

Topology

▶ **Definition 5** (Proj $\mathcal{A}$ as a set)**.** Proj $\mathcal{A}$ *is defined to be* $\{\mathfrak{p} \in \operatorname{Spec} A \mid \mathfrak{p}$ *is homogeneous and relevant*$\}$*, where*

- *an ideal* $\mathfrak{p} \subseteq A$ *is said to be* homogeneous *if for any* $a \in \mathfrak{p}$ *and* $i \in \mathbb{N}$*,* a_i *is in* $\mathfrak{p}$ *as well where* $a_i \in \mathcal{A}_i$ *is the* i*-th projection of* a *with respect to grading* $\mathcal{A}$*;*
- *an ideal* $\mathfrak{p} \subseteq A$ *is said to be* relevant *if* $\bigoplus_{i=1}^{\infty} \mathcal{A}_i \not\subseteq \mathfrak{p}$*.*

Similar to Spec construction in Section 2.2, there is a topology on Proj $\mathcal{A}$ whose close sets are exactly the zero loci where for any $s \subseteq A$, zero locus of s is $\{\mathfrak{p} \in \operatorname{Proj}\mathcal{A} \mid s \subseteq \mathfrak{p}\}$; this topology is also called the Zariski topology. For any $a \in A$, $D(a)$ denotes the set $\{x \in \operatorname{Proj}\mathcal{A} \mid a \notin x\}$.

▶ **Theorem 6.** *For any* $a \in A$*,* $D(a)$ *is open in Zariski topology and* $\{D(a) \mid a \in A\}$ *forms a basis of the Zariski topology.*

Proof. Proofs can be found in [14, 00JM] and [9, Chapter 2, proposition 2.5] ◀

Structure sheaf

Let $U \subseteq \operatorname{Proj}\mathcal{A}$ be an open subset. The *sections* on U are defined to be

$$\mathcal{O}(U) = \left\{ s \in \prod_{x\in U} A^0_x \mid s \text{ is locally a homogeneous fraction} \right\},$$

where $A^0_{\mathfrak{p}}$ denotes the homogeneous localization of A at a homogeneous prime ideal $\mathfrak{p}$, i.e. the subring of $A_{\mathfrak{p}}$ of elements of degree zero, and s is said to be *locally a homogeneous fraction* if for any $x \in U$, there is some open subset $x \in V \subseteq U$, $i \in \mathbb{N}$ and $a, b \in \mathcal{A}_i$ such that for all $y \in V$, $s(y) = \frac{a}{b}$. Equipped with the natural restriction maps, $\mathcal{O}$ defined in this way forms a presheaf; the sheaf condition of $\mathcal{O}$ is checked in the category of sets where it follows from the definition of locally homogeneous fractions. This sheaf is called the *s*tructure sheaf of Proj $\mathcal{A}$, also written as $\mathcal{O}_{\operatorname{Proj} A}$

Locally ringed spaces

▶ **Theorem 7.** *The stalk of* $(\operatorname{Proj}\mathcal{A},\mathcal{O})$ *at a homogeneous prime relevant ideal* $\mathfrak{p}$ *is isomorphic to* $A^0_{\mathfrak{p}}$.

Proof. It can be checked that the function $A^0_{\mathfrak{p}} \to \mathcal{O}_{\operatorname{Proj}\mathcal{A},\mathfrak{p}}$ defined by $\frac{a}{b} \mapsto \langle D(b), x \mapsto \frac{a}{b}\rangle$ is a ring isomorphism. Details can be found in [14, 01M4] ◀

Since $A^0_{\mathfrak{p}}$ is a local ring for any homogeneous prime ideal $\mathfrak{p}$, it can be concluded that $(\operatorname{Proj}\mathcal{A},\mathcal{O}_{\operatorname{Proj}\mathcal{A}})$ is a locally ringed space.

Affine cover

▶ **Lemma 8.** *For any* $x \in \operatorname{Proj}\mathcal{A}$*, there is some* $0 < m \in \mathbb{N}$ *and* $f \in \mathcal{A}_m$*, such that* $x \in D(f)$*, i.e.* $f \notin x$.

Proof. Let $x \in \operatorname{Proj}\mathcal{A}$, by construction, $\bigoplus_{i=1}^{\infty}\mathcal{A}_i \not\subseteq x$. Thus there is some $f = f_1 + f_2 + \cdots \notin x$, then at least one $f_i \notin x$ for otherwise $f \in x$. ◀

Thus, to construct an affine cover, it is sufficient to prove that for all $0 < m \in \mathbb{N}$ and homogeneous element $f \in \mathcal{A}_m$, $(D(f), \mathcal{O}_{\operatorname{Proj}\mathcal{A}}|_{D(f)})$ is isomorphic to $(\operatorname{Spec} A^0_f, \mathcal{O}_{\operatorname{Spec} A^0_f})$ where A^0_f is the subring of the localised ring A_f consisting of elements of degree zero. By fixing the previous notations, an isomorphism between locally ringed space is a pair (ϕ,α) where ϕ is a homeomorphism between the topological spaces $D(f)$ and $\operatorname{Spec} A^0_f$ and α an isomorphism between $\phi_*(\mathcal{O}_{\operatorname{Proj}\mathcal{A}}|_{D(f)})$ and $\mathcal{O}_{\operatorname{Spec} A^0_f}$.

▶ **Theorem 9.** $D(f) \cong \operatorname{Spec} A^0_f$ *are homeomorphic as topological spaces.*

The following proofs are an expansion of [9, II.2.5] while drawing ideas from [15, II.4.5].

Proof. Define $\phi : D(f) \to \operatorname{Spec} A^0_f$ by $\mathfrak{p} \mapsto \operatorname{span}\left\{\frac{g}{1} \mid g \in \mathfrak{p}\right\} \cap A^0_f$; by clearing denominators, one can show that $\phi(\mathfrak{p}) = \operatorname{span}\left\{\frac{g}{f^i} | g \in \mathfrak{p} \cap A_{mi}\right\}$. One can check that $\phi(\mathfrak{p})$ is indeed a prime ideal. ϕ is continuous by checking on the topological basis consisting of basic open sets of $\operatorname{Spec} A^0_f$. The fact that basic open sets form a basis is already recorded in mathlib. Take $\frac{a}{f^n} \in A^0_f$, then $\phi^{-1}(D(a/f^n)) = D(f) \cap D(a)$.

- $D(f) \cap D(a)$ is a subset of $\phi^{-1}(D(a/f^n))$ because if $y \in D(f) \cap D(a)$ and $a/f^n \in \phi(y)$, i.e. $a/f^n = \sum_i (c_i/f^{n_i})(g_i/1)$, then by multiplying suitable powers of f, $af^N/1 = (\sum_i c_i g_i f^{m_i})/1$ for some N, so by definition of localisation, $af^N f^M = \sum_i c_i g_i f^{m_i}$ for some M implying that $a \in y$. Contradiction.
- On the other hand, if $\phi(y) \in D(a/f^n)$ and $a \in y$, then $a/1 \in h(y)$, contradiction because $a/f^n = a/1 \, 1/f^n \in \phi(y)$.

For the other direction, define $\psi : \operatorname{Spec} A^0_f \to D(f)$ to be $x \mapsto \left\{a \mid \text{for all } i \in \mathbb{N}, \frac{a_i^m}{f^i} \in x\right\}$. For ψ to be well-defined, one needs to check that $\psi(x)$ is a homogeneous prime ideal that is relevant. Continuity of ψ depends on that ϕ and ψ are inverse to each other. $D(f)$ with the subspace topology has a basis of the form $D(f) \cap D(a)$, thus it is sufficient to prove that preimages of these sets are open. By considering $\phi(D(f) \cap D(a)) = \bigcup_i \phi(D(f) \cap D(a_i))$, each $\phi(D(f) \cap D(a_i))$ is open because $\phi(D(f) \cap D(a_i)) = D(a_i^m/f^i)$ in $\operatorname{Spec} A^0_f$. To prove $\phi(D(f) \cap D(a_i)) = D(a_i^m/f^i)$, it is sufficient to prove $\phi^{-1}(D(a_i^m/f^i)) = D(f) \cap D(a)$ and this is true by continuity of ϕ. Since ϕ and ψ are inverses to each other, preimage of $D(f) \cap D(a)$ is indeed $\phi(D(f) \cap D(a))$. ◀

Let ϕ and ψ be the continuous functions defined in the previous proof, U be an open subset of $\operatorname{Spec} A_f^0$, s be a section on $\phi^{-1}(U)$ and $x \in U$, then $\psi(x) \in \phi^{-1}(U)$, hence $s(\psi(x)) = \frac{n}{d} \in A^0_{\psi(x)}$ for some $i \in \mathbb{N}$ and $n, d \in \mathcal{A}_i$. Keeping the same notation, a ring homomorphism $\alpha_U : \phi_*(\mathcal{O}_{\mathrm{Proj}}\,|_{D(f)})(U) \to \mathcal{O}_{\operatorname{Spec} A_f^0}(U)$ can be defined as $s \mapsto \left(x \mapsto \frac{nd^{m-1}/f^i}{d^m/f^i}\right)$ where $n, d \in \mathcal{A}_i$. Assuming α_U is well-defined, it is easy to check that $U \mapsto \alpha_U$ is natural in U, hence α defines a morphism of sheaves.

▶ **Lemma 10.** *For any open subset $U \subseteq \operatorname{Spec} A_f^0$, α_U is well-defined; hence α defines a morphism of sheaves.*

Proof. It is clear that both the numerator and denominator have degree zero. Now $d^m/f^i \notin x$ follows from $d \notin \psi(x)$. Next $\alpha_U(s)$ is locally a fraction: since s is locally a quotient, for any $x \in U$, there is some open set $V \subseteq \mathrm{Proj}\mathcal{A}$ such that $\psi(x) \in V \subseteq \phi^{-1}(U)$ such that $s(y) = \frac{a}{b}$ for all $y \in V$ where $a, b \in A_n$ and $b \notin y$, then to check $\alpha_U(s)$ is locally quotient, use the open subset $\phi(V)$ and check that for all $z \in \phi(V)$, $\alpha_U(s)(z) = \frac{ab^{m-1}}{b^m}$. The proof of α_U being a ring homomorphism involves manipulations of fractions in localised rings, for more details, see Section 3. ◀

In the other direction, if $s \in \mathcal{O}_{\operatorname{Spec} A_f^0}(U)$ and $y \in \phi^{-1}(U)$, then $\phi(y) \in U$, so $s(\phi(y))$ can be written as $\frac{a}{b}$ where $a, b \in A_f^0$; then a can be written as $\frac{n_a}{f^{i_a}}$ for some $n_a \in A_{mi_a}$ and b as $\frac{n_b}{f^{i_b}}$ for some $n_b \in A_{mi_b}$. Hence, a ring homomorphism $\beta_U : \mathcal{O}_{\operatorname{Spec} A_f^0}(U) \to \mathcal{O}_{\mathrm{Proj}}|_{D(f)}\,(\phi^{-1}(U))$ can be defined as $s \mapsto \left(y \mapsto \frac{n_a f_b^i}{n_b f^{i_a}}\right)$. Assuming β is well defined, it is easy to check that the assignment $U \mapsto \beta_U$ is natural so that β is a natural transformation.

▶ **Lemma 11.** *For any open subset $U \subseteq \operatorname{Spec} A_f^0$, β_U is well-defined; hence β defines a morphism of sheaves.*

Proof. $n_a f_b^{i_b}$ and $n_b f^{i_a}$ have the same degree. $n_b f^{i_a} \notin y$ follows from $b \notin \phi(y)$. Since s locally is a fraction, there are open sets $\phi(y) \in V \subseteq U$, such that for all $z \in V$, $s(z)$ is $\frac{a/f^{l_1}}{b/f^{l_2}}$. Then on $\phi^{-1}(V) \subseteq \phi^{-1}(U)$, $\psi_U(s)(y)$ is always $\frac{af^{l_2}}{bf^{l_1}}$. Checking that β_U is a ring homomorphism involves manipulating fractions of fractions. ◀

▶ **Theorem 12.** *$\phi_*(\mathcal{O}_{\mathrm{Proj}\,\mathcal{A}}|_{D(f)})$ and $\mathcal{O}_{\operatorname{Spec} A_f^0}$ are isomorphic as sheaves.*

Proof. By combining Lemma 10 and Lemma 11, it is sufficient to check $\alpha \circ \beta$ and $\beta \circ \alpha$ are both identities.

- $\beta \circ \alpha = 1$: let $s \in \mathcal{O}_{\mathrm{Proj}}|_{D(f)}\,(\phi^{-1}(U))$, then for $x \in \phi^{-1}(U)$
$$\alpha_U(s) = x \mapsto \frac{nd^{m-1}/f^i}{d^m/f^i},$$
where $s(x) = \frac{n}{d}$. Thus, by definition
$$\beta_U(\alpha_U(s))(x) = \frac{nd^{m-1}f^i}{d^m f^i} = \frac{n}{d} = s(x).$$
- $\alpha \circ \beta = 1$: let $s \in \mathcal{O}_{\operatorname{Spec} A_f^0}(U)$, then for $x \in U$
$$\beta_U(s) = x \mapsto \frac{n_a f^{i_b}}{n_b f^{i_a}}$$
where $s(x) = \frac{n_a/f^{i_a}}{n_b/f^{i_b}}$. Thus
$$\phi_U(\psi_U(s))(x) = \frac{n_a f^{i_b}\left(n_b f^{i_a}\right)^{m-1}/f^j}{\left(n_b f^{i_a}\right)^m/f^j} = \frac{n_a/f^{i_a}}{n_b/f^{i_b}} = s(x).$$ ◀

▶ **Corollary 13.** *$(\mathrm{Proj}\,\mathcal{A}, \mathcal{O}_{\mathrm{Proj}\,\mathcal{A}})$ is a scheme.*

3 Formalisation details

3.1 Homogeneous Ideal

Let A be an R-algebra and an ι-grading $\mathcal{A} : \iota \to R$-submodules of A, `ideal.is_homogeneous` is the proposition of an ideal being homogeneous and `homogeneous_ideal` is the type of all homogeneous ideals of A [17]. Note that, by this implementation, homogeneous ideals are not literally ideals, for this reason, $\operatorname{Proj} \mathcal{A}$ cannot be implemented as a subset of $\operatorname{Spec} A$.

```
def ideal.is_homogeneous : Prop :=
∀ (i : ι) {|r : A|}, r ∈ I → (direct_sum.decompose 𝓐 r i : A) ∈ I

structure homogeneous_ideal extends submodule A A :=
(is_homogeneous' : ideal.is_homogeneous 𝓐 to_submodule)

def homogeneous_ideal.to_ideal (I : homogeneous_ideal 𝓐) : ideal A :=
    I.to_submodule

lemma homogeneous_ideal.is_homogeneous (I : homogeneous_ideal 𝓐) :
  I.to_ideal.is_homogeneous 𝓐 := I.is_homogeneous'

def homogeneous_ideal.irrelevant : homogeneous_ideal 𝓐 :=
⟨(graded_ring.proj_zero_ring_hom 𝓐).ker, omitted⟩
```

3.2 Homogeneous Localisation

If x is a multiplicatively closed subset of ring A, then the homogeneous localisation of A at x is defined to be the subring of localised ring A_x consisting of elements of degree zero. This ring is implemented as triples $\{(i, a, b) : \iota \times \mathcal{A}_i \times \mathcal{A}_i \mid b \notin x\}$ under the equivalence relation that $(i_1, a_1, b_1) \approx (i_2, a_2, b_2) \overset{\text{def}}{\iff} \frac{a_1}{b_1} = \frac{a_2}{b_2}$ in A_x. The quotient approach gives an induction principle via quotients, though the construction still uses classical reasoning, many lemmas will be automatic because of the rich API in mathlib about quotient spaces already; compared to the subring approach, one would need to write corresponding lemmas manually by excessively invoking `classical.some` and `classical.some_spec` which are APIs in Lean to extract the data and the corresponding proof from an existentially quantified proposition. One potential benefit of the subring approach is that different propositions can be specified for different multiplicative subsets to customize what properties and attributes are to be made explicit; for example for localisation away from a single element, it is useful to make powers of denominators explicit. But this would sacrifice a universal approach to homogeneous localisation for different multiplicative subsets so that auxiliary lemmas would have to be duplicated. To maintain consistency and prevent duplication, this paper will adopt the approach via quotient space. Before writing this paper, the subring approach has also been tested. Comparing the two approaches proves that there is no significant difference in the smoothness of two formalisations but the quotient approach has a smaller code size.

```
variables {ι R A: Type*} [add_comm_monoid ι] [decidable_eq ι]
variables [comm_ring R] [comm_ring A] [algebra R A]
variables (𝓐 : ι → submodule R A) [graded_algebra 𝓐]
variables (x : submonoid A)

structure num_denom_same_deg :=
(deg : ι) (num denom : 𝓐 deg) (denom_mem : (denom : A) ∈ x)
```

```
def embedding (p : num_denom_same_deg 𝒜 x) : localization x :=
localization.mk p.num ⟨p.denom, p.denom_mem⟩

def homogeneous_localization : Type* := quotient (setoid.ker $ embedding 𝒜 x)
```

Then if `(y : homogeneous_localization 𝒜 x)`, its value, degree, numerator and denominator can all be defined by using induction/recursion principles for quotient spaces:

```
variable (y : homogeneous_localization 𝒜 x)

def val : localization x :=
  quotient.lift_on' y (num_denom_same_deg.embedding 𝒜 x) $ λ _ _, id

def num : A := (quotient.out' y).num
def denom : A := (quotient.out' y).denom
def deg : ι := (quotient.out' y).deg

lemma denom_mem : y.denom ∈ x := (quotient.out' y).denom_mem
lemma num_mem_deg : y.num ∈ 𝒜 f.deg := (quotient.out' y).num.2
lemma denom_mem_deg : y.denom ∈ 𝒜 y.deg := (quotient.out' y).denom.2
lemma eq_num_div_denom : y.val = localization.mk y.num ⟨y.denom, y.denom_mem⟩ :=
omitted
```

3.3 The Zariski Topology

In this section A will be graded by $\mathbb{N}$ and the grading denoted by $\mathcal{A}$. $\operatorname{Proj} \mathcal{A}$ is formalised a structure:

```
structure projective_spectrum :=
(as_homogeneous_ideal : homogeneous_ideal 𝒜)
(is_prime : as_homogeneous_ideal.to_ideal.is_prime)
(not_irrelevant_le : ¬(homogeneous_ideal.irrelevant 𝒜 ≤ as_homogeneous_ideal))
```

After building more API around `projective_spectrum`, the Zariski topology with a basis of basic open sets can be formalised as:

```
def zero_locus (s : set A) : set (projective_spectrum 𝒜) :=
{x | s ⊆ x.as_homogeneous_ideal}

instance zariski_topology : topological_space (projective_spectrum 𝒜) :=
topological_space.of_closed (set.range (zero_locus 𝒜)) omitted omitted omitted

def basic_open (r : A) : topological_space.opens (projective_spectrum 𝒜) :=
{ val := { x | r ∉ x.as_homogeneous_ideal },
  property := ⟨{r}, set.ext $ λ x, set.singleton_subset_iff.trans $ not_not.symm⟩ }

lemma is_topological_basis_basic_opens : topological_space.is_topological_basis
  (set.range (λ (r : A), (basic_open 𝒜 r : set (projective_spectrum 𝒜)))) :=
omitted
```

3.4 Locally Ringed Spaces

mathlib provides `Top.presheaf.is_sheaf_iff_is_sheaf_comp` to check the sheaf condition by composing a forgetful functor and `Top.subsheaf_to_Types` to construct subsheaf of types satisfying a local predicate [6]; $\mathcal{O}_{\mathrm{Spec}}$ in mathlib adopted this approach [5], and structure sheaf of Proj will also be constructed in this way. `is_locally_fraction` is a local predicate expressing "being locally a homogeneous fraction" in Section 2.3:

```
def is_fraction {U : opens (Proj 𝒜)} (f : Π x : U,  A_x^0 ) : Prop :=
∃ (i : ℕ) (r s : 𝒜 i), ∀ x : U, ∃ (s_nin : s.1 ∉ x.1.as_homogeneous_ideal),
  f x = quotient.mk' ⟨i, r, s, s_nin⟩

def is_fraction_prelocal : prelocal_predicate (λ (x : Proj 𝒜),  A_x^0 ) :=
{ pred := λ U f, is_fraction f,
  res := by rintros V U i f ⟨j, r, s, w⟩; exact ⟨j, r, s, λ y, w (i y)⟩ }

def is_locally_fraction : local_predicate (λ (x : Proj 𝒜),  A_x^0 ) :=
(is_fraction_prelocal 𝒜).sheafify

def structure_sheaf_in_Type : sheaf Type* (Proj 𝒜):=
subsheaf_to_Types (is_locally_fraction 𝒜)
```

The presheaf of rings is also defined as `structure_presheaf_in_CommRing` and it is checked that composition with the forgetful functor is naturally isomorphic to the underlying presheaf of `structure_sheaf_in_Type` which implies that `structure_presheaf_in_CommRing` satisfies the sheaf condition as well by using `Top.presheaf.is_sheaf_iff_is_sheaf_comp`.

```
def structure_presheaf_in_CommRing : presheaf CommRing (Proj 𝒜) :=
{ obj := λ U, CommRing.of ((structure_sheaf_in_Type 𝒜).1.obj U), ..omitted }

def structure_presheaf_comp_forget :
  structure_presheaf_in_CommRing 𝒜 ⋙ (forget CommRing) ≅
  (structure_sheaf_in_Type 𝒜).1 :=
omitted

def Proj.structure_sheaf : sheaf CommRing (Proj 𝒜) :=
⟨structure_presheaf_in_CommRing 𝒜, (is_sheaf_iff_is_sheaf_comp _ _).mpr
  (is_sheaf_of_iso (structure_presheaf_comp_forget 𝒜).symm
    (structure_sheaf_in_Type 𝒜).cond)⟩
```

Then following Theorem 7, `stalk_to_fiber_ring_hom` is a family of ring homomorphism $\prod_x \mathcal{O}_{\mathrm{Proj}\,\mathcal{A},x} \to A^0_x$ obtained by universal property of colimit with its right inverse as a family of function `homogeneous_localization_to_stalk`:

```
def stalk_to_fiber_ring_hom (x : Proj 𝒜) :
  (Proj.structure_sheaf 𝒜).presheaf.stalk x ⟶ CommRing.of A_x^0 :=
limits.colimit.desc (((open_nhds.inclusion x).op) ⋙ (Proj.structure_sheaf 𝒜).1)
  omitted

def section_in_basic_open (x : Proj 𝒜) :
  Π (f : A_x^0), (Proj.structure_sheaf 𝒜).1.obj (op (Proj.basic_open 𝒜 f.denom)) :=
λ f, ⟨λ y, quotient.mk' ⟨_, ⟨f.num, _⟩, ⟨f.denom,_⟩, _⟩, _⟩

def homogeneous_localization_to_stalk (x : Proj 𝒜) :
  A_x^0 → (Proj.structure_sheaf 𝒜).presheaf.stalk x :=
λ f, (Proj.structure_sheaf 𝒜).presheaf.germ
  (⟨x, homogeneous_localization.mem_basic_open _ x f⟩ : Proj.basic_open _ f.denom)
  (section_in_basic_open _ x f)
```

```
def Proj.stalk_iso' (x : Proj 𝒜) :
  (Proj.structure_sheaf 𝒜).presheaf.stalk x ≃+* CommRing.of A^0_x :=
ring_equiv.of_bijective (stalk_to_fiber_ring_hom _ x)
  ⟨omitted, function.surjective_iff_has_right_inverse.mpr
    ⟨homogeneous_localization_to_stalk 𝒜 x, omitted⟩⟩
```

Hence establishing that $\operatorname{Proj} \mathcal{A}$ is a locally ringed space:

```
def Proj.to_LocallyRingedSpace : LocallyRingedSpace :=
{ local_ring := λ x, @@ring_equiv.local_ring _
    (show local_ring  A^0_x , from infer_instance) _
    (Proj.stalk_iso' 𝒜 x).symm,
    ..(Proj.to_SheafedSpace 𝒜) }
```

3.5 Affine cover

```
variables {f : A} {m : ℕ} (f_deg : f ∈ 𝒜 m) (x : Proj| D(f))
```

`Spec.T` and `Proj.T` denote the topological space associated with each locally ringed space. Let $0 < m \in \mathbb{N}$ and $f \in \mathcal{A}_m$ and $x \in D(f)$, by following Theorem 9, the continuous function ϕ is formalised as `Proj_iso_Spec_Top_component.to_Spec` where continuity is checked on basic open sets:

```
namespace Proj_iso_Spec_Top_component
namespace to_Spec

def carrier : ideal A^0_f :=
ideal.comap (algebra_map A^0_f A_f)
  (ideal.span $ algebra_map A (away f) '' x.val.as_homogeneous_ideal)

def to_fun : Proj.T| D(f) → Spec.T A^0_f :=
λ x, ⟨carrier 𝒜 x, omitted /-a proof for primeness-/⟩

end to_Spec

def to_Spec (f : A) : Proj.T| D(f) ⟶ Spec.T A_f :=
{ to_fun := to_Spec.to_fun 𝒜 f,
  continuous_to_fun := omitted }
```

Similarly, ψ is defined as a function first, then the fact that ϕ and ψ are inverses to each other is formalised next as `to_Spec_from_Spec` and `from_Spec_to_Spec` respectively. The continuity of ψ hence follows.

```
namespace from_Spec

def carrier (q : Spec.T A^0_f) : set A :=
{a | ∀ i, (quotient.mk' ⟨_, ⟨proj 𝒜 i a ^ m, _⟩, ⟨f^i, _⟩, _⟩ : A^0_f) ∈ q.1}

def carrier.as_ideal : ideal A := { carrier := carrier f_deg q, ..omitted }
def carrier.as_homogeneous_ideal : homogeneous_ideal 𝒜 :=
⟨carrier.as_ideal f_deg hm q, omitted⟩

def to_fun : Spec.T A^0_f → Proj.T| D(f) :=
λ q, ⟨⟨carrier.as_homogeneous_ideal f_deg hm q, omitted, omitted⟩, omitted⟩
```

```

end from_Spec

lemma to_Spec_from_Spec : to_Spec.to_fun 𝒜 f (from_Spec.to_fun f_deg hm x) = x :=
omitted
lemma from_Spec_to_Spec : from_Spec.to_fun f_deg hm (to_Spec.to_fun 𝒜 f x) = x :=
omitted

def from_Spec : Spec.T A^0_f ⟶ Proj.T| D(f) :=
{ to_fun := from_Spec.to_fun f_deg hm,
  continuous_to_fun := omitted }

end Proj_iso_Spec_Top_component
```

The homeomorphism between $D(f)$ and Spec A^0_f is achieved by combining ϕ and ψ together.

```
def Proj_iso_Spec_Top_component:
  Proj.T| D(f) ≅ Spec.T (A^0_f) :=
{ hom := Proj_iso_Spec_Top_component.to_Spec 𝒜 f,
  inv := Proj_iso_Spec_Top_component.from_Spec hm f_deg,
  ..omitted /-composition being identity-/ }
```

Then by following Lemma 11, β is formalised as `Proj_iso_Spec_Sheaf_component.from_Spec`.

```
namespace Proj_iso_Spec_Sheaf_component
namespace from_Spec
```

Let V be an open set in Spec A^0_f and s be a section on V, then let y be an element of $\phi^{-1}(V)$,

```
variables (V : (opens (Spec A^0_f))^op) (s : (Spec A^0_f).presheaf.obj V)
variables (y : ((@opens.open_embedding Proj.T D(f)).is_open_map.functor.op.obj
  ((opens.map (Proj_iso_Spec_Top_component hm f_deg).hom).op.obj V)).unop)
-- This is but a verbose way of spelling y is in ϕ^{-1}(V) for type checking reasons.
```

one can evaluate $s(\phi(y))$ and represent the result as a fraction $\frac{a}{b}$ where $a = \frac{n_a}{f^{i_a}}$ and $b = \frac{n_b}{f^{i_b}}$.

```
-- Corresponding to evaluating a section in Lemma 11.s(ϕ(y))
def data : structure_sheaf.localizations A^0_f
  ((Proj_iso_Spec_Top_component hm f_deg).hom ⟨y.1, _⟩) :=
s.1 ⟨_, _⟩

-- s(ϕ(y)) = a/b, this is a, see Lemma 11.
def data.num : A^0_f := omitted

-- s(ϕ(y)) = a/b, this is b, see Lemma 11
def data.denom : A^0_f := omitted
```

Then $\frac{n_a f^{i_b}}{n_b f^{i_a}}$ is a homogeneous fraction in A^0_y. The function thus defined is indeed a ring homomorphism and locally a fraction. This sheaf morphism is recorded as `from_Spec` where its naturality is checked automatically by Lean's simplifier.

```
-- s ↦ (y ↦ n_a f^{i_b}/n_b f^{i_a}), this is n_a f^{i_b}, see Lemma 11.
def num : A :=
  (data.num _ hm f_deg s y).num * (data.denom _ hm f_deg s y).denom

-- s ↦ (y ↦ n_a f^{i_b}/n_b f^{i_a}), this is n_b f^{i_a}, see Lemma 11.
```

```
def denom : A :=
  (data.denom _ hm f_deg s y).num * (data.num _ hm f_deg s y).denom

def bmk : A^0_y :=
quotient.mk'
{ deg := (data.num _ hm f_deg s y).deg + (data.denom _ hm f_deg s y).deg,
  num := ⟨num hm f_deg s y, _⟩,
  denom := ⟨denom hm f_deg s y, _⟩,
  denom_mem := omitted }

def to_fun.aux : ((Proj_iso_Spec_Top_component hm f_deg).hom _* (Proj|
    D(f)).presheaf).obj V :=
⟨bmk hm f_deg V s, omitted /-being locally a homogeneous fraction-/⟩

def to_fun : (Spec A^0_f).presheaf.obj V ⟶
  ((Proj_iso_Spec_Top_component hm f_deg).hom _* (Proj| D(f)).presheaf).obj V :=
{ to_fun := λ s, to_fun.aux 𝒜 hm f_deg V s, ..omitted /-ring homomorphism
    proofs-/ }

end from_Spec

def from_Spec : (Spec A^0_f).presheaf ⟶
  (Proj_iso_Spec_Top_component hm f_deg).hom _* (Proj| D(f)).presheaf :=
{ app := λ V, from_Spec.to_fun 𝒜 hm f_deg V,
  naturality' := λ _ _ _, by { ext1, simp } }

end Proj_iso_Spec_Sheaf_component
```

By following Lemma 10, α is formalised as `Proj_iso_Spec_Sheaf_component.to_Spec`: let U be an open set in $\operatorname{Spec} A^0_f$ and s a section in $\phi_*(\mathcal{O}_{\mathrm{Proj}}|_{D(f)})(U)$, then let y be any point in U,

```
namespace Proj_iso_Spec_Sheaf_component
namespace to_Spec
variable (U : (opens (Spec.T A^0_f))^op)
variable (s : ((Proj_iso_Spec_Top_component hm f_deg).hom _*
    (Proj| D(f))).presheaf.obj U) -- (ϕ_*(𝒪_Proj|_D(f)))(U)
```

After evaluating $s(\psi(y))$, the result can be represented as $\frac{n}{d}$ where n, d both have degree i. Then $\frac{nd^{m-1}}{f^i}$ and $\frac{d^m}{f^i}$ are both homogeneous fractions of the same degree and hence $({nd^{m-1}}/{f^i})/({d^m}/{f^i})$ is an element of the twice localised ring $\left(A^0_f\right)_y$. The function thus defined is a ring homomorphism and locally a fraction. This sheaf morphism is recorded as `to_Spec` where its naturality is checked automatically by Lean's simplifier.

```
-- evaluating a section, this is s(ψ(y))
def hl (y : unop U) : homogeneous_localization 𝒜 _ :=
s.1 ⟨((Proj_iso_Spec_Top_component hm f_deg).inv y.1).1, _⟩

-- s ↦ (x ↦ (nd^{m-1}/f^i)/(d^m/f^i)) where n,d ∈ 𝒜_i, this is nd^{m-1}/f^i, see Lemma 10.
def num (y : unop U) : A^0_f :=
quotient.mk'
{ deg := m * (hl hm f_deg s y).deg,
  num := ⟨(hl hm f_deg s y).num * (hl hm f_deg s y).denom ^ m.pred, _⟩,
  denom := ⟨f^(hl hm f_deg s y).deg, _⟩,
```

```
  denom_mem := _ }

def denom (y : unop U) : A_f^0 :=
quotient.mk'
{ deg := m * (hl hm f_deg s y).deg,
  num := ⟨(hl hm f_deg s y).denom ^ m, _⟩,
  denom := ⟨f ^ (hl hm f_deg s y).deg,_⟩,
  denom_mem := _ }

def fmk (y : unop U) : (A_f^0)_y :=
mk (num hm f_deg s y) ⟨denom hm f_deg s y, _⟩

def to_fun :
  ((Proj_iso_Spec_Top_component hm f_deg).hom _* (Proj| D(f))).obj U ⟶
  (Spec A_f^0).presheaf.obj U :=
{ to_fun := λ s, ⟨λ y, fmk hm f_deg s y, omitted /-proof of being locally a
    fraction-/⟩, ..omitted /-proof of being a ring homomorphism-/},
end to_Spec

def to_Spec :
  (Proj_iso_Spec_Top_component hm f_deg).hom _* (Proj| D(f)).presheaf ⟶
  (Spec A_f^0).presheaf :=
{ app := λ U, to_Spec.to_fun hm f_deg U,
  naturality' := λ U V subset1, by { ext1, simp } }
end Proj_iso_Spec_Sheaf_component
```

After checking `from_Spec` (β) and `to_Spec` (α) compose to identity, one establishes that $(D(f), \mathcal{O}_{\operatorname{Proj} \mathcal{A}})$ is isomorphic $(\operatorname{Spec} A_f^0, \mathcal{O}_{\operatorname{Spec} A_f^0})$ as locally ringed spaces. Hence $\operatorname{Proj} \mathcal{A}$ with structure sheaf , $\mathcal{O}_{\operatorname{Proj} \mathcal{A}}$ is a scheme.

```
def Sheaf_component:
  (Proj_iso_Spec_Top_component hm f_deg).hom _* (Proj| D(f)).presheaf ≅
  (Spec A_f^0).presheaf :=
{ hom := Proj_iso_Spec_Sheaf_component.to_Spec 𝒜 hm f_deg,
  inv := Proj_iso_Spec_Sheaf_component.from_Spec 𝒜 hm f_deg,
  ..omitted /-composition is identity-/ }

def iso:
  (Proj| D(f)) ≅ Spec A_f^0 :=
let H : (Proj| D(f)).to_PresheafedSpace ≅ (Spec A_f^0).to_PresheafedSpace :=
  PresheafedSpace.iso_of_components
    (Proj_iso_Spec_Top_component hm f_deg) (Sheaf_component 𝒜 f_deg hm) in
LocallyRingedSpace.iso_of_SheafedSpace_iso
{ hom := H.1, inv := H.2, hom_inv_id' := H.3, inv_hom_id' := H.4 }

def Proj.to_Scheme : Scheme :=
{ local_affine := omitted,..Proj }
```

This concludes the formalisation of the Proj construction for any $\mathbb{N}$-graded rings. In [17], $R[X_0, \ldots, X_n]$ is endowed with a grading by its R-submodule of homogeneous polynomials of fixed degrees so that projective n-space over R can be formalised as `Proj.to_Scheme (λ i, mv_polynomial.homogeneous_submodule (fin (n + 1)) R i)`; similarly, once the fact that quotient operation induces a grading on the quotiented object is formalised, projective varieties can also be defined using `Proj.to_Scheme`.

3.6 Reflections on the formalisation

An example of a calculation

Most calculations in proofs of Theorem 9 and Lemmas 10 and 11 are omitted. I present the details of verifying β_U preserves multiplication to showcase the flavour of calculations involved. Verifying that β_U preserving zero and one is similar but slightly simpler while preservation of addition is more cumbersome. Since α only involves one layer of fractions, calculations are not as long.

Let x, y be two sections, the aim is to show $\beta_U(xy) = \beta_U(x)\beta_U(y)$, i.e. for all $z \in \phi^{-1}(U)$, $\beta_U(xy)(z) = \beta_U(x)(z)\beta_U(y)(z)$.

```
lemma bmk_mul (x y : (Spec A⁰_f).presheaf.obj V) :
  bmk hm f_deg V (x * y) = bmk hm f_deg V x * bmk hm f_deg V y :=
begin
  ext1 z,
```

by writing $x(\phi(z))$ as $\frac{a_x/f^{i_x}}{b_x/f^{j_x}}$, $y(\phi(z))$ as $\frac{a_y/f^{i_y}}{b_y/f^{j_y}}$ and $(xy)(\phi(z)) = \frac{a_{xy}/f^{i_{xy}}}{b_{xy}/f^{j_{xy}}}$, one deduces that $\frac{a_x a_y/f^{i_x+i_y}}{b_x b_y/f^{j_x+j_y}} = \frac{a_{xy}/f^{i_{xy}}}{b_{xy}/f^{j_{xy}}}$, by definition of equality in localised ring, it implies that, there is some $\frac{c}{f^l}$ such that

$$\frac{a_x a_y b_{xy} c}{f^{i_x+i_y+j_{xy}+l}} = \frac{a_{xy} b_x b_y c}{f^{i_{xy}+j_x+j_y+l}}.$$

```
  have mul_eq := data.eq_num_div_denom hm f_deg (x * y) z,
  ... -- simplification
  erw is_localization.eq at mul_eq,
  obtain ⟨⟨C, hC⟩, mul_eq⟩ := mul_eq, -- C is the c/f^l above.
  ...

  -- c ∉ z
  have C_not_mem : C.num ∉ z.1.as_homogeneous_ideal := omitted,

  -- setting up notations.
  set a_xy := _, set i_xy := _, set b_xy := _, set j_xy := _,
  set a_x := _, set i_x := _, set b_x := _, set j_x := _,
  set a_y := _, set i_y := _, set b_y := _, set j_y := _,
  set l := _,
  ...
```

By definition of equality in localisation again, there exists some $n_1 \in \mathbb{N}$ such that

$$a_x a_y b_{xy} c f^{i_x y+j_x+j_y+l+n_1} = a_{xy} b_x b_y c f^{i_x+i_y+j_{xy}+l+n_1} \tag{1}$$

```
  obtain ⟨⟨_, ⟨n1, rfl⟩⟩, mul_eq⟩ := mul_eq,
```

The aim is to show

$$\frac{a_{xy} f^{j_{xy}}}{b^{xy} f^{i_x}} = \frac{a_x f^{j_x}}{b^x f^{i_x}} \frac{a_y f^{j_y}}{b_y f^{i_y}},$$

by Equation (1) and definition of equality in localised ring, cf^{l+n_1} verifies this equality.

```
suffices : (mk (a_xy * f ^ j_xy) ⟨b_xy * f ^ i_xy, _⟩ : localization.at_prime _)
= mk (a_x * f ^ j_x) ⟨b_x * f ^ i_x, _⟩ * mk (a_y * f ^ j_y) ⟨b_y * f ^ i_y, _⟩
  := omitted,
  ...
refine ⟨⟨C.num * f^(1 + n1), _⟩, _⟩,
  ...
end
```

In totality, this is about ~100 lines of code by following essentially three lines of calculation when done with pen-and-paper. Admittedly, the above code is not the most optimal, but the magnitude is not greatly exaggerated. Strictly speaking, setting 13 variable names takes a lot of code and is not necessary, but with readable variable names, rewriting is made much simpler in the latter stage of this calculation. I think the following factors contribute to the differences between formalisation and a pen-and-paper-proof:

- Every element of a localised ring can be written as a fraction of a numerator and a denominator is a corollary of the construction but does not follow straightly from its definition. When written on a paper, it is often read "let $\frac{a}{b} \in A_{\mathfrak{p}}$" while in Lean it is becomes `intro x, set x_denom := ..., set x_num := ..., have eq1 : x.val = x_num / x_denom := ...`. This problem is more noticeable when `rewrite [eq1]` is unsound. Thus, many extra steps are required to set up the proof.
- Elements of a (homogeneously) localised ring contain not only data, but proofs as well. For example, the denominator of an element is a term `⟨d, some_proof⟩` of a subtype. This makes `rewrite` less smooth to use, for equalities are often of the form `h : d = d'`, thus `rewrite [h]` is type theoretically unsound.
- Terms of `localization x` or `homogeneous_localization 𝒜 x` have to contain proofs to make the definitions correct, thus constructing any term of these types requires many proofs or disproofs of membership. Thus, a formalised calculation cannot be as liberal as a pen-and-paper-proof when come to whether the terms are well-defined. The situation can be partially mitigated by writing a simple tactic to try lemmas involving degrees of an element in a graded object, for example automatically splitting `a * b ∈ 𝒜 (m + n)` to `a ∈ 𝒜 (m + n)` to `a ∈ 𝒜 m` and `b ∈ 𝒜 n` and try recursively try to solve both. However, if non-definitional equalities is involved, tactics would be less helpful, when the subterms are in the wrong order, one needs to manually re-organise the subterms into its correct order to use the customary tactic.
- Not many high powered tactics are available for localised ring, for example `ring` will be able to solve `x * y = y * x` and much more complicated goal in a commutative ring, but `ring` cannot (and should not be able to) solve `(a / b * c / d : localization _) = c / b * a / d`.

The first three bullet points are essentially all because formalisation requires more rigour than that of pen-and-paper proofs; whether the requirement of extra rigour is beneficial is another question and not in the scope of this paper. For the fourth bullet point, it is definitely helpful to have a tactic automating many proofs, the catch is that equality in localised ring is existentially quantified – $\frac{a}{b} = \frac{a'}{b'}$ if and only if $ab'c = a'bc$ for some c in a multiplicative subset, while proving $ab'c = a'bc$ is easily mechanized by the `ring` tactic, providing c to Lean is certainly hard to be made trivial by any tactic soon. Thus, a tactic can only do so much without human input for now.

On propositional equality

Originally, I expected propositional equalities that are not equal by definition such as $\phi(\psi(y)) = y$ in Theorem 9 would pose a challenge, but the difficulty is less severe: indeed, I only need to prove some redundant lemma like $\phi(\psi(y))$ is in some open sets that clearly contains y; the reason is that in this project I did not compare algebraic structures depending on propositional equality, i.e. $\mathcal{O}_y$ and $\mathcal{O}_{\phi(\psi(y))}$; but foreseeably, this difficulty will come back when one starts to develop the theory of projective variety furtherer.

4 Conclusion

Since a large part of modern algebraic geometry depends on the Proj construction, much potential future research is possible: calculating cohomology of projective spaces; defining projective morphisms; Serre's twisting sheaves to name a few. Other approaches to the Proj construction also exist, for example, by gluing a family of schemes together; however, since there is no other formalisation of the Proj construction, I could not compare different approaches or compare capabilities of formalising modern algebraic geometry of different theorem provers. Thus I would like to conclude this paper with an invitation/challenge – state and formalise something involving more than affine schemes in your preferred theorem prover; for the only way to know which, if any, theorem provers handle modern mathematics satisfactorily is to actually formalise more modern mathematics.

References

1. Anthony Bordg, Lawrence Paulson, and Wenda Li. Simple type theory is not too simple: Grothendieck's schemes without dependent types. *Experimental Mathematics*, 31(2):364–382, 2022. `doi:10.1080/10586458.2022.2062073`.
2. Kevin Buzzard, Chris Hughes, Kenny Lau, Amelia Livingston, Ramon Fernández Mir, and Scott Morrison. Schemes in Lean. *Experimental Mathematics*, 31(2):355–363, 2022.
3. Laurent Chicli. *Une formalisation des faisceaux et des schémas affines en théorie des types avec Coq*. PhD thesis, INRIA, 2001.
4. Wei-Liang Chow. On compact complex analytic varieties. *American Journal of Mathematics*, 71(4):893–914, 1949. URL: `http://www.jstor.org/stable/2372375`.
5. Mathlib Contributors. Lean mathlib. `https://github.com/leanprover-community/mathlib`, 2022.
6. Mathlib Contributors. Mathlib documentation. `https://leanprover-community.github.io/mathlib_docs`, 2023.
7. Leonardo de Moura, Soonho Kong, Jeremy Avigad, Floris Van Doorn, and Jakob von Raumer. The Lean theorem prover (system description). In *International Conference on Automated Deduction*, pages 378–388. Springer, 2015.
8. Roger Godement. Topologie algébrique et théorie des faisceaux. *Publications de*, 1, 1958.
9. Robin Hartshorne. Graduate texts in mathematics. *Algebraic Geometry*, 52, 1977.
10. Saunders MacLane and Ieke Moerdijk. *Sheaves in geometry and logic: A first introduction to topos theory*. Springer Science & Business Media, 2012.
11. Anders Mörtberg and Max Zeuner. Towards a formalization of affine schemes in cubical agda.
12. Amnon Neeman. *Algebraic and analytic geometry*. Cambridge University Press, 2007.
13. Jean-Pierre Serre. Géométrie analytique et géométrie algébrique. *Ann. Inst. Fourier, VI (1955–56)*, pages 1–42, 1955.
14. The Stacks Project Authors. *Stacks Project*. `https://stacks.math.columbia.edu`, 2018.
15. Ravi Vakil. The rising sea: Foundations of algebraic geometry. 2017. *URL http://virtualmath1.stanford. edu/˜ vakil/216blog*, 24:29, 2017.

16 Vladimir Voevodsky, Benedikt Ahrens, Daniel Grayson, et al. UniMath — a computer-checked library of univalent mathematics. available at `http://unimath.org`. URL: `https://github.com/UniMath/UniMath`.

17 Eric Wieser and Jujian Zhang. Graded rings in Lean's dependent type theory. In *International Conference on Intelligent Computer Mathematics*, pages 122–137. Springer, 2022.