Coq による三角形三色問題の証明

橋本 翔太　木村 大輔

Coq とは数学の証明作成を支援する定理証明支援系である．人間は Coq と対話的に証明作成を行うことで誤りを排除した信頼できる証明を得ることができる．三角形三色問題とは，n 段の逆三角形に配置された六角形のマスに対して，互いに隣接する任意の 3 マスの色が全て同じかどれも異なるように逆三角形を 3 色で塗り分けたとき，逆三角形の 3 頂点のマスの色が必ず全て同じ，もしくはどれも異なるような段数の一般項を求める問題である．この問題は雑誌「数学セミナー」の「エレガントな解答もとむ」欄に出題されており，一般項は 3^k 段の形で表せることが示されている．本研究では，数学セミナーでの証明を Coq で形式化して証明を完成させた． これにより，幾何的な直観に頼った側面のあった元の議論を論理に基づいた形式的な証明に直すことができた.

1 はじめに

Coq [1] とは数学の定理や補題，主張の正しさを保証するためのソフトウェアの 1 つである．証明の作成中の各場面で示すべき主張（サブゴール）に対して人間がサブゴールを示すための次の一手を指示すると Coq は次のサブゴールを提示し，人間の次の一手を待つ．このような対話的なやりとりにより Coq は証明の完成の手助けをする．こういったソフトウェアを定理証明支援系と呼ぶ．証明の規模が大きくなると，複雑な場合分けの漏れがあったり計算ミスなど機械的操作のミスにより人間は誤った証明をしてしまうことがある．Coq の支援を受けることで，このような誤りが排除された信頼できる証明を得ることができる．また，Coq はプログラミング言語でもあるため，作成した証明の複製が容易である．Coq を用いて作成した証明ファイルを公開することで，多くの人がこれらを保証済みの証明として各々の目的達成のために利用することができる．

A formal proof for the three-colored triangle problem on Coq

Shota Hashimoto, 東邦大学大学院理学研究科, Toho University.

Daisuke Kimura, 東邦大学大学院理学研究科, Toho University.

本研究では，Coq + SSReflect を用いて三角形三色問題の証明の形式化を完成させた．†1 SSReflect [2] [3] は「証明と計算は区別でき，計算は証明を要求しない」(ポワンカレ原理) のポリシーに基づいて設計された Coq の拡張ライブラリである．つまり，証明によるリーズニングよりも計算 (等式変形) を積極的に用いた方が証明を簡略化できる．簡単な同値変形で示すことができる命題論理や等式・不等式に関する主張の証明などは論理式として推論で示すよりも bool 型の項と見なして変形した方が効率がよく，SSReflect はその機能を提供する．

三角形三色問題 [4] [5] [6] とは，次のような問題である：n 段の逆三角形に配置された正六角形のすべてのマスを異なる 3 色を用いて色分けをする．ただし，隣り合う 2 マスとそれらに接する下の段のマスの色は，どれも同じかどれも異なるように塗り分ける．このとき，逆三角形の段数が 3, 9, 27 段の場合 †2 は，規則に従ったどのような色の塗り方をしても逆三角形の端点の 3 マスの色はどれも同じかどれも異なる

†1 `https://github.com/SyotaHashimoto/ThreeColor` から完成させた Coq のコード (“Coq による三角形三色問題の証明.v”) をダウンロードすることができる．

†2 本稿では 0 段目, 1 段目, 2 段目,. . . と数える.「n 段の逆三角形」と書いた場合，一辺のマスの個数は $n + 1$ 個である．

が，一般にはこのような性質は成り立たない．このような性質を満たす段数の一般項は何か？という問題である．この問題の解決方法は大きく分けて 3 つの方法が知られている．[4]

1. パスカルの三角形の値を mod 3 としたものをマスの色と対応させて使い，代数学での $Lucas$ の定理と関係のある命題を示すことで解決する方法．
2. 逆三角形のマスの塗り方が n 個の独立パターンに分解できること利用して，重ね合わせの原理を用いることで解決する方法．
3. 比較的少ない段数で成立する段数を調べて段数の規則性 (数列) を見つけ出し，この数列から予測できる段数の一般項を推測する方法．

3 の方法において一般項が 3^k 段であると推測されており，推測された一般項の段数でないならば逆三角形の塗り方の規則に従わない反例が存在することも知られている．

本研究では 3. の方法で推測して得られた一般項が必要十分条件になっていることを Coq+SSReflect を用いて証明した．Coq に実装するにあたって，三角形三色問題は幾何的な側面を多くもつ問題であるためこのままでは Coq にコードとして実装することができない．そこで，三角形のマスの状況を表現する論理式を用意し，色塗り規則を関数化することで三角形三色問題の状況を形式化した．また，十分条件の証明の方法としては一般項に現れる自然数 k に関する数学的帰納法を用いて証明した．一方で，必要条件は対偶法と n について場合分けをすることで証明した．

本論文の第 2 章では三角形三色問題の証明の概要について述べる．第 3 章では三角形三色問題の証明を Coq に実装するために必要な準備について述べる．第 4 章では実際に Coq に実装した三角形三色問題の証明について述べる．第 5 章では三角形三色問題を形式化することで得られた知見について述べる．第 6 章ではまとめを述べる．

2　三角形三色問題の概要

三角形三色問題について述べる前に調和性と調和彩色三角形の定義について先に述べる．以下，3 色のどれかの色が塗られた同じ大きさの正六角形のマスが平面上に逆三角形の形に敷き詰められている状況を考える (図 3 参照) †3.

定義 2.1 (調和性)**.** (隣接しているとは限らない) 3 つのマスに塗られている色がすべて同じか相異なるとき，この 3 マスは**調和性を満たす**，または，**調和している** という．

例 2.2. 図 1 のような 3 マスの組は調和性を満たしているが，図 2 のような 3 マスの組は調和性を満たしていない．

図 1　調和性を満たす 3 色の組

図 2　調和性を満たさない 3 色の組

定義 2.3 (彩色三角形)**.** 隣接した全ての 3 マスが調和性を満たすように色が塗られた逆三角形を**彩色三角形**という．

定義 2.4 (調和彩色三角形)**.** 3 つの端点のマスに塗られている色が調和性を満たしている彩色三角形を**調和彩色三角形** ($WellColoredTriangle$) という．

次に三角形三色問題について述べる．$n(>0)$ 段の彩色三角形がある．図 3 は $n=9$ のときの彩色三角形である．このとき，最下段のマスの色は赤であり，最上段の両端のマスは黄，青であるから最上段の両端のマスの色と最下段のマスの色について調和性を満たしている (つまり，調和彩色三角形である) ことが分かる．

$n=9$ の場合は最上段の塗り方が図 3 の塗り方でなくとも逆三角形の 3 つの頂点のマスは調和性を満

†3　塗られる色は青，赤，黄を想定しているが，図の中では紙面の都合を考慮してそれぞれ黒，灰色，白に変換している．

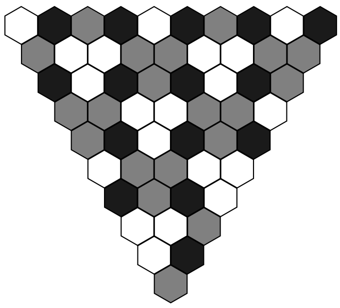

図 3　彩色三角形（$n = 9$ のとき）

たすことが観察できる．このことから次のような仮説が考えられる．

(仮説)　最上段をどのように塗っても最上段の両端のマスの色と最下段のマスの色は調和性を満たす．

数学セミナー誌で出題された三角形三色問題は次の 2 つの問題のことである [5].

1. $n = 9$ のとき仮説が成立することを証明せよ．
2. $n = 9$ 以外に仮説が成立する段数が存在するか調べ，存在するならば n の一般式を求めよ．

この問題については既に解答が得られており，一般に $n = 3^k$ 段の逆三角形において仮説が成立することを示す次の定理 2.5 が示されている．

定理 2.5. $n(> 0)$ 段の逆三角形に配置されたマス対して，最上段のマスを 3 色で任意に塗ったとき $(\exists k.n = 3^k) \Leftrightarrow n$ 段の逆三角形は常に調和彩色三角形．

本節ではこの定理の証明を Coq で実装するにあたり，証明の概要について述べる．

2.1　十分条件

定理 2.5 を証明するにあたって，十分条件と必要条件に分けて話を進める．ここでは定理 2.5 の十分条件である補題 2.6 の証明の概要について述べる．

補題 2.6 (十分条件)**.** n 段の逆三角形に配置されたマス対して，

$(\exists k.n = 3^k) \Rightarrow n$ 段の逆三角形は常に調和彩色三角形．

補題 2.6 を証明するためには論理同値である次の命題を証明すればよい．

$\forall k.(n = 3^k \Rightarrow n$ 段の逆三角形は常に調和彩色三角形$)$．

これは k に関する数学的帰納法を用いて証明する．

- $k = 0$ のときは $n = 1$ となり明らかに成立する．
- k のとき成立すると仮定する．すなわち，3^k 段の逆三角形ならば常に調和彩色三角形であると仮定する．

ここからは図 4 を用いて証明を進める．図 4 では各マスに塗られている色を表している．ただし，図中の c_y^x は左から x 番目，上から y 段目のマスに塗られている色を表している． ここで，図 4

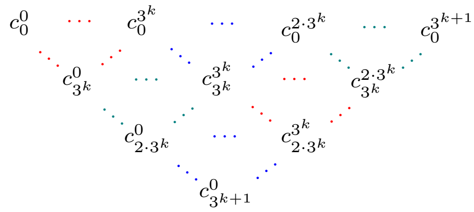

図 4　彩色三角形（$n = 3^{k+1}$ のとき）

の中にあるいくつかの 3^k 段の彩色三角形に注目する．注目する彩色三角形のそれぞれを，その 3 つの頂点のマスの色を組にして表すことにする．上から 0 段目から 3^k 段目の間では次の 3 つの組で表現される逆三角形に注目する．

$$\left(c_0^0, c_0^{3^k}, c_{3^k}^0\right),\quad \left(c_0^{3^k}, c_0^{2\cdot 3^k}, c_{3^k}^{3^k}\right),$$
$$\left(c_0^{2\cdot 3^k}, c_0^{3^{k+1}}, c_{3^k}^{2\cdot 3^k}\right).$$

また，上から 3^k 段目から $2 \cdot 3^k$ 段目の間にある彩色三角形は次の 2 個に注目する．

$$\left(c_{3^k}^0, c_{3^k}^{3^k}, c_{2\cdot 3^k}^0\right),\quad \left(c_{3^k}^{3^k}, c_{3^k}^{2\cdot 3^k}, c_{2\cdot 3^k}^{3^k}\right).$$

さらに，上から $2 \cdot 3^k$ 段目から 3^{k+1} 段目の間にある彩色三角形は次の 1 個に注目する．

$$\left(c_{2\cdot 3^k}^0, c_{2\cdot 3^k}^{3^k}, c_{3^{k+1}}^0\right).$$

これらの 6 個の 3^k 段の彩色三角形はすべて帰納法の仮定より常に調和彩色三角形である．よって，調和彩色三角形の定義より最上段のマスの 4 色 c_0^0，$c_0^{3^k}$，$c_0^{2\cdot 3^k}$，$c_0^{3^{k+1}}$ から最下段のマスの $c_{3^{k+1}}^0$ の色が得られる．最後に，4 色 c_0^0，$c_0^{3^k}$，$c_0^{2\cdot 3^k}$，$c_0^{3^{k+1}}$ から得られた色は $c_0^0, c_0^{3^{k+1}}$ が規則に従った色と等しくなること †4 を利用すると，最上段の両端のマスに塗られている 2 色 c_0^0，$c_0^{3^{k+1}}$ と最下段の色 $c_{3^{k+1}}^0$ は調和性を満たす．すなわち，

†4　$3^4 = 81$ 通りの色の場合分けをすることで示すことができるが，詳細は補題 3.11 で述べる．

3^{k+1} 段の逆三角形は常に調和彩色三角形である．

2.2 必要条件

次は定理 2.5 の必要条件である補題 2.7 の証明の概要について述べる．

補題 2.7 (必要条件)**.** $n(>0)$ 段の逆三角形に配置されたマス対して，

n 段の逆三角形は常に調和彩色三角形 $\Rightarrow (\exists k.n = 3^k)$.

補題 2.7 の証明では対偶法を用いた後に，n に関する場合分けをして証明する．補題 2.7 の対偶は以下の通り．

$\neg$ ($\exists$ $k.n = 3^k$) $\Rightarrow$ $\neg$(n 段の逆三角形は常に調和彩色三角形).

したがって，$\neg(\exists k.n = 3^k)$ を仮定したとき，次の各場合について調和彩色三角形にならない最上段のマスの塗り方を挙げればよい．場合分けの仕方は次の 3 つである．

1. n が偶数
2. n が奇数 かつ $3^k < n \leq 2 \cdot 3^k$
3. n が奇数 かつ $2 \cdot 3^k + 1 \leq n < 3^{k+1}$

各場合について調和彩色三角形にならないような最上段のマスの塗り方は次の通り．

- 1. のときは最上段のマスを黄，青の順で交互に塗る．(図 5 参照) すると，黄，青を交互に塗られているので第 1 段目のマスは規則よりすべて赤色で塗られている．さらに，もう一方の規則より最下段のマスまですべて赤で塗られている．したがって，n が偶数であるから最上段の両端のマスは黄であり，最下段のマスの色が赤であるから調和性を満たさないので調和彩色三角形でない．

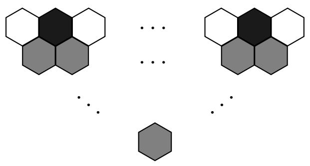

図 5　n が偶数

- 2. のときは最上段のマスを外側の両端のマスから内側の方に向かって黄，青の順で 2 色を用いて対称的に交互に塗る．(図 6 参照) すると，補題 2.6 より 3^k 段の逆三角形は調和彩色三角形なので，最上段から 3^k 段下のマスは黄，青の順で交互に塗られている．これは 1. の場合に帰着できるので最下段のマスの色は赤である．したがって，最上段の両端のマスの色は黄であり，最下段のマスの色が赤であるから調和性を満たさないので調和彩色三角形でない．

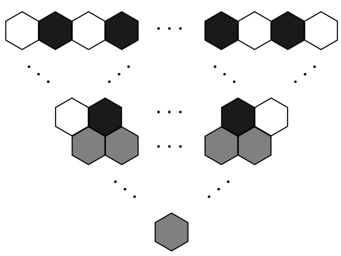

図 6　n が奇数 かつ $3^k < n \leq 2 \cdot 3^k$

- 3. のときは最上段のマスを両端のマスからそれぞれ 3^k マス内側の方に向かって黄，その他の内側のマスを青を用いて対称的に塗る．(図 7 参照) このとき，黄で塗られている内側のマスは $n - 2 \cdot 3^k + 1$ マスである．すると，補題 2.6 より 3^k 段の逆三角形は調和彩色三角形なので，最上段から 3^k 段において外側から $n - 2 \cdot 3^k + 1$ マスはすべて赤で塗られている．同様にして，$2 \cdot 3^k$ 段下のマスの色はすべて赤で塗られている．したがって，最上段の両端のマスの色は黄であり，最下段のマスの色が赤であるから調和性を満たさないので調和彩色三角形でない．

3　Coq に実装するための準備

3.1　調和彩色三角形の定義

三角形三色問題のような幾何的な直観に基づく問題や前節で述べたような証明を Coq で形式化するには，問題の暗黙の前提や色塗り規則などを観察して適切な形式化の方法を選ぶ必要がある．この問題では各マスにはただ一色だけ塗られることが暗黙の前提としてあるため，マスの場所からそのマスの色を返す関数を用いることにする．また，この問題では最上段の色を

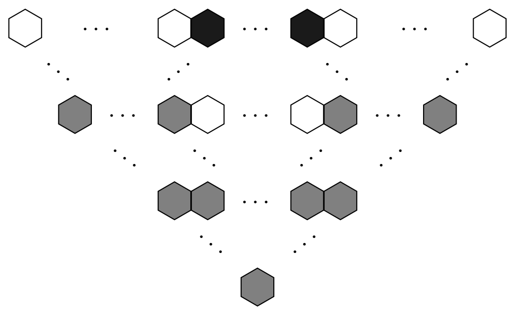

図 7　n **が奇数 かつ** $2 \cdot 3^k + 1 \leq n < 3^{k+1}$

指定すれば色塗り規則に従えばその下の色も帰納的に求められるので，最上段の色塗りを与える関数から全体の色塗りを与える関数へ拡張できる．これらの観察結果が今回の Coq 上での形式化のアイデアである．

まず，実装する際に用いた定義や関数，論理式について述べる．以下，$f : A \to B \to C$ のような型をもつ項と $a : A$ と $b : B$ について説明のために $f(a, b)$ と記述することがあるが，これは fab と同義である．

定義 3.1 ($Color$)**.** マスに塗る色の集合を次のように定義する．

$$Color \stackrel{def}{=} \{red, yel, blu\}.$$

このとき，red は red，yel は yellow，blu は blue を表している．以降，red を r，yel は y，blu は b として略記することもある．

定義 3.2 (mix)**.** $mix : Color \to Color \to Color$ を以下で定義する．

$$(r, r) \mapsto r, \quad (r, y) \mapsto b, \quad (r, b) \mapsto y,$$
$$(y, r) \mapsto b, \quad (y, y) \mapsto y, \quad (y, b) \mapsto r,$$
$$(b, r) \mapsto y, \quad (b, y) \mapsto r, \quad (b, b) \mapsto b.$$

演算 mix は塗り方の規則を再現するための関数の 1 つである．引数となる 2 色が同じ場合は同じ色を返し，異なる場合は 2 色とも異なる第三の色を返す関数である．

定義 3.3 (彩色関数)**.** マスの場所からそのマスの色を返す型 $\mathbb{N} \to \mathbb{N} \to Color$ の関数を**彩色関数**と呼ぶ．本稿では $cpos$ を彩色関数を意味する変数として用いることにし，$cpos$ の型を明記せずに省略することもある．$cpos(x, y)$ は左から x 番目，上から y 段目のマスの色を意味する．

例 3.4. 図 3 を与える彩色関数 $cpos$ は，逆三角形の 3 つの端点のマスに関して次を満たす：

- $cpos(0, 0) = yel$,
- $cpos(9, 0) = blu$,
- $cpos(0, 9) = red$.

定義 3.5 ($Fmix$)**.** $cpos : \mathbb{N} \to \mathbb{N} \to Color$ に対して，$Fmix(cpos)$ を次のように定義する．

$Fmix(cpos) \stackrel{def}{\Longleftrightarrow} (\forall\ x, y \in \mathbb{N},\ (cpos(x, y) = mix\ (cpos(x, y),\ cpos(x + 1, y)).$

$Fmix$ は互いに隣接する任意の 3 つのマスは調和性を満たしていることを論理式に書き直したものである．すなわち，逆三角形に塗られている色はランダムに塗られているわけではなく，x, y を任意にすることで規則に従ってマスの色が塗られた彩色三角形であることを表している．

定義 3.6 ($WellColoredTriangleF$)**.** $x, n \in \mathbb{N}$ に対して，$WellColoredTriangleF(x, n)$ を次のように定義する：

$WellColoredTriangleF(x, n) \stackrel{def}{\Longleftrightarrow} (\forall cpos : \mathbb{N} \to \mathbb{N} \to Color, (Fmix(cpos) \to TriangleF(cpos, x, 0, n)).$

ただし，$TriangleF(cpos, x, y, n)$ は以下で定義されている．

$TriangleF(cpos, x, y, n) \stackrel{def}{\Longleftrightarrow} cpos(x, y + n) = mix(cpos(x, y), cpos(x + n, y)).$

$WellColoredTriangleF$ は定義 2.4 で述べた n 段の調和彩色三角形の定義を最上段の色の塗り方の関数 $cpos$ を用いて論理式に書き直したものである．x は逆三角形の左端のマス $(x, 0)$ †5 を基準として定めるために用いており，n は逆三角形の一辺の長さを表している．よって，三角形三色問題におけるマスの塗り方に従っていれば，彩色三角形の端点の 3 つのマス $(x, 0), (x + n, 0), (x, n)$ に塗られている色は調和性を満たしていることを表している．

3.2　彩色条件

ここからはマスに塗る色の塗り方を表す関数について述べていく．まず，最初に必要条件で用いる最上

†5　左から x 番目，上から y 段目のマスを座標のように (x, y) と表している．すなわち，$(x, 0)$ は左から x 番目，上から 0 段目 (最上段) のマスを表している．

段のマスの塗り方を 3 つ紹介する．

定義 3.7 ($colorYB$)**.** $colorYB : \mathbb{N} \to \mathbb{N} \to \mathbb{N} \to Color$ を以下で定義する．

$$colorYB(x,n,z) \stackrel{def}{=} \begin{cases} yel & (0 \le z-x \le n \wedge z-x \text{ が奇数}) \\ blu & (0 \le z-x \le n \wedge z-x \text{ が偶数}) \\ blu & (\text{otherwise}) \end{cases}$$

$colorYB$ は補題 2.7 の証明において n が偶数のときの最上段のマスの塗り方を表した関数である．一番左端にあるマスを基準（左から x 番目のマス）としたときに，基準から右に z 番目のマスを指定するときには $z-x$ を用いて指定している．$z-x$ は基準となるマスから離れているマス数を表しており，相対的に最上段のマスを指定している．また，$colorYB$ は最上段のマスを交互に塗っていることを $z-x$ の偶奇によって定めている．

定義 3.8 ($colorYBBY$)**.** $colorYBBY : \mathbb{N} \to \mathbb{N} \to \mathbb{N} \to Color$ を以下で定義する．

$$colorYBBY(x,n,z) \stackrel{def}{=} \begin{cases} yel & (0 \le z-x \le n/2 \wedge z-x \text{ が偶数}) \\ yel & (n/2+1 \le z-x \le n \wedge z-x \text{ が奇数}) \\ blu & (0 \le z-x \le n/2 \wedge z-x \text{ が奇数}) \\ blu & (n/2+1 \le z-x \le n \wedge z-x \text{ が偶数}) \\ yel & (\text{otherwise}) \end{cases}$$

$colorYBBY$ は補題 2.7 の証明において n が奇数 かつ $3^k < n \le 2 \cdot 3^k$ のときの最上段のマスの塗り方を表した関数である．$colorYB$ と同様にして基準となるマスから離れているマス数を用いて相対的に最上段のマスを 1 つ指定している．また，$0 \le z-x \le n/2$ の範囲では左端のマスから偶数番目のときは yel，奇数番目のときは blu を塗り，$n/2+1 \le z-x \le n$ の範囲では塗る色が入れ替わる．このようにして，$colorYBBY$ は外側から内側に向かって対称的に最上段のマスを交互に塗っていることを $z-x$ の偶奇によって定めている．

定義 3.9 ($colorBYB$)**.** $colorBYB : \mathbb{N} \to \mathbb{N} \to \mathbb{N} \to \mathbb{N} \to Color$ を以下で定義する．

$$colorBYB(x,n,k,z) \stackrel{def}{=} \begin{cases} yel & (3^k \le z-x \le n-3^k) \\ blu & (\text{otherwise}) \end{cases}$$

$colorBYB$ は補題 2.7 の証明において n が奇数かつ $2 \cdot 3^{k'} + 1 \le n < 3^{k'+1}$ のときの最上段のマスの塗り方を表した関数である．$colorYBBY$ は基準となるマスから右に 3^k 番目から $n-3^k$ 番目までマスの色を黄色で塗り，その他を青で塗ることで再現している．

ここまで，最上段の色の塗り方を関数で表すことができた．次は最上段のマスの塗り方 (関数) を全体の色を決める彩色関数に拡張する関数を定義する．

定義 3.10 ($liftpaint$)**.** $liftpaint : (\mathbb{N} \to Color) \to \mathbb{N} \to \mathbb{N} \to Color$ を以下で再帰的関数として定義する．

$$liftpaint(topcol,x,y) \stackrel{def}{=} \begin{cases} topcol(x) & (y=0 \text{ のとき}) \\ mix(liftpaint(topcol,x,y'), liftpaint(topcol,x+1,y')) & \\ & (y=y'+1 \text{ のとき}) \end{cases}$$

演算 $liftpaint$ は最上段の色塗りを与える関数 $topcol$ から全体の色塗りを与える彩色関数へ拡張する．$y=0$ のときは最上段のマスは最上段の色塗り関数 $topcol$ に従って塗っていることを表している．また，$y=y'+1$ の形をしているときは，最上段から y' 段目にある隣り合う 2 マスの色に対して，mix を適用することでその間にある y 段目のマスの色が得られることを表している．

3.3 関数の性質

次に証明を円滑に進めていくために用いた関数の性質に関する補題について述べる．

補題 3.11 ($mixCut$)**.** $\forall c_0, c_1, c_2, c_3 \in Color$ に対して，$mix(mix(mix(c_0,c_1), mix(c_1,c_2)), mix(mix(c_1,c_2), mix(c_2,c_3))) = mix(c_0,c_3)$.

$mixCut$ は演算 mix のもつ性質を論理式にしたものであり，mix と 4 色を用いて表された色は 2 色のみを用いて書き換えることができることを表している．証明する際には各色が 3 通りずつ取り得るので合計 $3^4 = 81$ 通りの場合分けをおこなって mix の計算をすれば証明することができる．三角形三色問題の十分

条件 (補題 2.6) を証明する際に用いる補題である．

補題 3.12 (*AllRed*)**.** $\forall cpos : \mathbb{N} \to \mathbb{N} \to Color$, $x, y, n \in \mathbb{N}$ に対して，$Fmix(cpos) \Rightarrow (\forall i. \in \mathbb{N}, (0 \le i \le n \Rightarrow cpos(x+i, y) = red)) \Rightarrow cpos(x, y+n) = red$.

三角形三色問題の必要条件 (補題 2.7) の証明において，n がどの場合でもすべてのマスが赤に塗られている段があることに帰着させて矛盾を導いている．AllRed により，すべてのマスが赤で塗られている段があるときは最下段のマスは赤であることがいえる．

補題 3.13 (*cposF*)**.** $\forall topcol : \mathbb{N} \to Color$ に対して，$Fmix(liftpaint(topcol))$.

cposF は $liftpaint(topcol)$ に従ってマスの色を塗ると，常に *Fmix* を満たすことを表している．すなわち，最上段のマスの塗り方を表す関数 *topcol* が何であってもこれを拡張した彩色関数 $liftpaint(topcol)$ は互いに隣接する任意の 3 マスは調和性を満たすこと表している．

4 Coq における三角形三色問題

4.1 十分条件

補題 2.6 を Coq に実装するために論理式の形にしたものが次の定理 4.1 である．

定理 4.1 (十分条件)**.** $\forall x, n \in \mathbb{N}, (\exists k \in \mathbb{N}, n = 3^k \Rightarrow WellColoredTriangleF(x, n))$.

証明．補題 2.6 (pp.3) でも述べたように次を証明することで，定理 4.1 を示す．

$\forall cpos, \forall k, n, x, y \in \mathbb{N},\ n = 3^k \Rightarrow (Fmix(cpos) \Rightarrow TriangleF(cpos, x, y))$.

これを k に関する数学的帰納法を用いて証明する．

$k = 0$ のときは $n = 1$ となるので明らかに成立する．

次に k のとき成立すると仮定して $k+1$ のときも成立することを示す．

3^k マスずつ離れたマスの色を図 8 のように表すことにする． このとき，3 つの端点のマスの色が $c_0 = cpos(x, y^k)$，$c_3 = cpos(x, y)$，$c_5 = cpos(x+3^k, y^k)$ である 3^k 段の彩色三角形が調和彩色三角形であることは，帰納法の仮定からすぐに示せる．同様にして，この彩色三角形以外にも 5 つの彩色三角形は調和彩色三角形である．すなわち，次の 6 つが成立する．

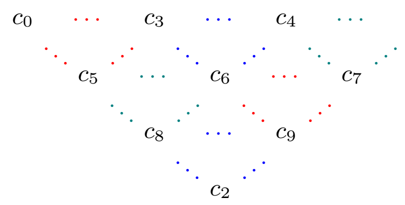

図 8　彩色三角形（$n = 3^{k+1}$ のとき）

- $TriangleF(cpos, x, y, 3^k)$
- $TriangleF(cpos, x+3^k, y, 3^k)$
- $TriangleF(cpos, x+2\cdot 3^k, y, 3^k)$
- $TriangleF(cpos, x, y+3^k, 3^k)$
- $TriangleF(cpos, x+3^k, y+3^k, 3^k)$
- $TriangleF(cpos, x, y+2\cdot 3^k, 3^k)$

これらと補題 3.11 (*mixCut*) を用いて式変形をすると，$cpos(x, y+3^k) = mix(cpos(x, y), cpos(x+3^{k+1}, y))$ が得られる．すなわち，$TriangleF(cpos, x, y, 3^{k+1})$．したがって，$TriangleF(cpos, x, y, n)$ である． □

4.2 必要条件

補題 2.7 を Coq に実装するために論理式の形にしたものが次の定理 4.2 である．

定理 4.2 (必要条件)**.** $\forall x, n \in \mathbb{N}, n > 0 \Rightarrow (WellColoredTriangleF(x, n) \Rightarrow \exists k \in, n = 3^k)$

補題 2.7 (pp.4) でも述べたように次の対偶を証明することで定理 4.2 を示す．

$\forall n, x \in \mathbb{N}, n > 0 \Rightarrow (\neg(\exists k \in \mathbb{N}, n = 3^k) \Rightarrow \neg WellColoredTriangleF(x, n))$

この対偶を証明するためには，

- $n > 0$
- $\neg(\exists k \in \mathbb{N}, n = 3^k)$,
- $WellColoredTriangleF(x, n)$

を仮定して矛盾を示せばよい．今回は n に関する場合分けをしてから各場合において矛盾を導く．

4.2.1 n が偶数の場合

n が偶数のときは補題 4.3，4.4 を証明してから，補題 4.5 を証明して矛盾を導く．

補題 4.3 (*EvenA*)**.** $\forall cpos, \forall x, n \in \mathbb{N}, n > 0 \Rightarrow Fmix(cpos) \Rightarrow (\forall i \in \mathbb{N}, (0 \le i \le n \Rightarrow cpos(x +$

$i, 0) = colorYB(x, n, x+i))) \Rightarrow (\forall i \in \mathbb{N}, (0 \le i \le n-1 \Rightarrow cpos(x+i, 1) = red))$.

補題 4.3 は最上段のマスの色を関数 $colorYB$ で塗ると，最上段より 1 段下の段のマスの色はすべて赤であることを表している．

証明．$0 \le i \le n-1$ を満たす i を任意にとると，仮定より $cpos(x+i, 0) = colorYB(x, n, x+i)$, $cpos(x+i+1, 0) = colorYB(x, n, x+i+1)$．また，$Fmix(cpos)$ より $cpos(x+i, 1) = mix(cpos(x+i, 0), cpos(x+i+1, 0))$ が導ける．

- i が偶数のとき
  $colorYB$ の定義より，$colorYB(x, n, x+i) = blu$, $colorYB(x, n, x+i+1) = yel$ であるから $cpos(x+i, 1) = red$.
- i が奇数のとき
  $colorYB$ の定義より，$colorYB(x, n, x+i) = yel$, $colorYB(x, n, x+i+1) = blu$ であるから $cpos(x+i, 1) = red$.

よって，i の遇奇にかかわらず $cpos(x+i, 1) = red$. □

補題 4.4 ($EvenB$)**.** $\forall cpos, \forall x, n \in \mathbb{N}, n > 0 \Rightarrow Fmix(cpos) \Rightarrow (\forall i \in \mathbb{N}, (0 \le i \le n \Rightarrow cpos(x+i, 0) = colorYB(x, n, x+i))) \Rightarrow (cpos(x, n) = red)$.

補題 4.4 は最上段のマスの色を関数 $colorYB$ で塗ると，最下段のマスの色は赤になるということを表している．

証明．補題 4.3 より $\forall i \in \mathbb{N}, (0 \le i \le n-1 \Rightarrow cpos(x+i, 1) = red)$．さらに，補題 3.12 より $cpos(x, n) = red$. □

補題 4.5 ($Even$)**.** $\forall x, n \in \mathbb{N}, (n > 0 \wedge odd(n) = false) \Rightarrow \neg WellColoredTriangleF(x, n)$.

ただし，補題 4.5 の中にある $odd(n)$ は次のように $SSReflect$ で定義されている関数である．

自然数 n に対して，

$$odd(n) \stackrel{def}{=} \begin{cases} true & (n \text{ が奇数}) \\ false & (otherwise) \end{cases}$$

証明．補題 3.13 より $\exists cposYB, Fmix(cposYB) \wedge \forall x_1, y_1 \in \mathbb{N}, cposYB(x_1, y_1) = liftpaint(colorYB(x, n), x_1, y_1)$．さらに，存在する $cposYB$ をそのまま $cposYB$ として名付けると，$\forall i \in \mathbb{N}, colorYB(x, n, x+i) = cposYB(x+i, 0)$ を満たす．また，$0 \le 0 \le n$, $0 \le n \le n$ を満たすので $colorYB(x, n, x) = colorYB(x, n, x+n) = yel$. さらに，仮定より $TriangleF(cpos, x, 0, n)$ であるから $cposYB(x, n) = yel$ となる．一方で，補題 4.4 より $cpos(x, n) = red$ となるので矛盾する． □

4.2.2 n が奇数 かつ $3^k < n \le 2 \cdot 3^k$ の場合

n が奇数 かつ $3^{k'} < n \le 2 \cdot 3^k$ のときは補題 4.6, 4.7, 4.8 を証明してから，補題 4.9 を証明して矛盾を導く．

補題 4.6 ($ShortOddA$)**.** $\forall cpos, \forall x, n, k \in \mathbb{N}, (3^k < n \le (3^k \cdot 2) \wedge odd(n) = true) \Rightarrow n > 0 \Rightarrow Fmix(cpos) \Rightarrow (\forall x_1, y_1 \in \mathbb{N}, TriangleF(cpos, x_1, y_1, 3^k)) \Rightarrow (\forall i \in \mathbb{N}, (0 \le i \le n \Rightarrow cpos(x+i, 0) = colorYBBY(x, n, x+i))) \Rightarrow (\forall i \in \mathbb{N}, (0 \le i \le n-3^k \Rightarrow cpos(x+i, 3^k) = colorYB(x, n-3^k, x+i)))$.

補題 4.6 は最上段のマスの色を関数 $colorYBBY$ で塗ると，最上段より 3^k 下の段のマスは黄，青で交互に塗ってあることを表している．

証明．$0 \le i \le n-3^k$ を満たす i を任意にとると，$0 \le i \le n$, $0 \le i+3^k \le n$ であるから仮定より，$cpos(x+i, 0) = colorYBBY(x, n, x+i)$, $cpos(x+i+3^k, 0) = colorYBBY(x, n, x+i+3^k))$．また，仮定の $TriangleF(cpos, x+i, 0, 3^k)$ より $cpos(x+i, 3^k) = mix(cpos(x+i, n), cpos(x+i+3^k, 0))$ が成立する．さらに，n は奇数であり $0 \le i \le n/2$, $n/2+1 \le i+3^k \le n$ を満たすので $colorYBBY$, $colorYB$ の色は i の遇奇によって定まる．

- i が偶数のとき
  $colorYBBY$ の定義より $colorYBBY(x, n, x+i) = yel$, $colorYBBY(x, n, x+i+3^k) = yel$ であり，$colorYB$ の定義より $colorYB(x, n-3^k, x+i) = yel$. よって，$cpos(x+i, 3^k) = mix(yel, yel) = yel = colorYB(x, n-3^k, x+i)$.
- i が奇数のとき

$colorYBBY$ の定義より $colorYB(x, n, x+i) = blu$，$colorYB(x, n, x+i+3^k) = blu$ であり，$colorYB$ の定義より $colorYB(x, n-3^k, x+i) = blu$ よって，$cpos(x+i, 3^k) = mix(blu, blu) = blu = colorYB(x, n-3^k, x+i)$.

以上より，i の遇奇にかかわらず $cpos(x+i, 3^k) = colorYB(x, n-3^k, x+i))$. □

補題 4.7 ($ShortOddB$)**.** $\forall cpos, \forall x, n, k \in \mathbb{N}, (3^k < n \leq 3^k \cdot 2 \wedge odd(n) = true) \Rightarrow n > 0 \Rightarrow Fmix(cpos) \Rightarrow (\forall x_1, y_1 \in \mathbb{N}, TriangleF(cpos, x_1, y_1, 3^k)) \Rightarrow (\forall i \in \mathbb{N}, (0 \leq i \leq n \Rightarrow cpos(x+i, 0) = colorYBBY(x, n, x+i))) \Rightarrow (\forall i \in \mathbb{N}, (0 \leq i \leq n-3^k-1 \Rightarrow cpos(x+i, 3^k+1) = red))$.

補題 4.7 は最上段のマスの色を関数 $colorYBBY$ で塗ると，最上段から 3^k+1 下の段のマスはすべて赤であるということを表している.

証明. 補題 4.6 より $\forall i \in \mathbb{N}, (0 \leq i \leq n-3^k \Rightarrow cpos(x+i, 3^k) = colorYB(x, n-3^k, x+i))$ となるので，$cpos(x+i, 3^k) = colorYB(x, n-3^k, x+i)$, $cpos(x+i+1, 3^k) = colorYB(x, n-3^k, x+i+1)$. $Fmix(cpos)$ より $cpos(x+i, 3^k+1) = mix(cpos(x+i, 3^k), cpos(x+i+1, 3^k))$. ここで，補題 4.6 と同様にして i の偶奇で場合分けをする.

- i が偶数のとき
  $colorYB$ の定義より $colorYB(x, n-3^k, x+i) = yel$，$colorYB(x, n-3^k, x+i+1) = blu$. よって，$cpos(x+i, 3^k+1) = mix(cpos(x+i, 3^k), cpos(x+i+1, 3^k)) = mix(yel, blu) = red$.
- i が奇数のとき
  $colorYB$ の定義より $colorYB(x, n-3^k, x+i) = blu$，$colorYB(x, n-3^k, x+i+1) = yel$. よって，$cpos(x+i, 3^k+1) = mix(cpos(x+i, 3^k), cpos(x+i+1, 3^k)) = mix(blu, yel) = red$.

以上より，i の遇奇にかかわらず $cpos(x+i, 3^k+1) = red$. □

補題 4.8 ($ShortOddC$)**.** $\forall cpos, \forall x, n, k \in \mathbb{N}, (3^k < n \leq 3^k \cdot 2 \wedge odd(n) = true) \Rightarrow n > 0 \Rightarrow Fmix(cpos) \Rightarrow (\forall x_1, y_1 \in \mathbb{N}, TriangleF(cpos, x_1, y_1, 3^k)) \Rightarrow (\forall i \in \mathbb{N}, (0 \leq i \leq n \Rightarrow cpos(x+i, 0) = colorYBBY(x, n, x+i))) \Rightarrow (\forall i \in \mathbb{N}, (0 \leq i \leq n-3^k-1 \Rightarrow cpos(x, n) = red))$.

補題 4.8 は最上段のマスの色を関数 $colorYBBY$ で塗ると最下段のマスの色は赤になることを表している.

証明. 補題 4.7 より $\forall i \in \mathbb{N}, (0 \leq i \leq n-3^k-1 \Rightarrow cpos(x+i, 3^k+1) = red)$. さらに，補題 3.12 より $cpos(x, n) = red$. □

補題 4.9 ($ShortOdd$)**.** $\forall x, n, k \in \mathbb{N}, (3^k < n \leq 3^k \cdot 2 \wedge odd(n) = true) \Rightarrow \neg WellColoredTriangleF(x, n)$.

証明. 補題 3.13 より $\exists cposYBBY, Fmix(cposYBBY) \wedge \forall x_1, y_1 \in \mathbb{N}, cposYBBY(x_1, y_1) = liftpaint(colorYBBY(x, n), x_1, y_1)$. さらに，存在する $cposYBBY$ をそのまま $cposYBBY$ として名付けると，$\forall i \in \mathbb{N}, colorYBBY(x, n, x+i) = cposYBBY(x+i, 0)$ を満たす. これより $colorYBBY(x, n, x) = cposYBBY(x, 0)$，$colorYBBY(x, n, x+n) = cposYBBY(x+n, 0)$. n が奇数であるから $colorYBBY(x, n, x) = colorYBBY(x, n, x+n)$ が成立するので，$colorYBBY(x, n, x) = colorYBBY(x, n, x+n) = yel$. さらに，仮定より $TriangleF(cposYBBY, x, 0, n)$ であるから $cposYBBY(x, n) = yel$ となる. 一方で，定理 4.1，補題 4.8 より $cpos(x, n) = red$ となるので矛盾する. □

4.2.3 n が奇数 かつ $2 \cdot 3^k + 1 \leq n < 3^{k+1}$ の場合

n が奇数 かつ $2 \cdot 3^k + 1 \leq n < 3^{k+1}$ のときは補題 4.10，4.11，4.12 を証明してから，補題 4.13 を証明して矛盾を導く.

補題 4.10 ($LongOddA$)**.** $\forall cpos, \forall x, n, k \in \mathbb{N}, (3^k \cdot 2+1 \leq n < 3^{k+1}) \Rightarrow Fmix(cpos) \Rightarrow (\forall x_1, y_1 \in \mathbb{N}, TriangleF(cpos, x_1, y_1, 3^k)) \Rightarrow (\forall i \in \mathbb{N}, (0 \leq i \leq n \Rightarrow cpos(x+i, 0) = colorBYB(x, n, x+i))) \Rightarrow ((\forall i \in \mathbb{N}, (0 \leq i \leq n-3^k \cdot 2 \Rightarrow cpos(x+i, 3^k) = red)) \wedge (\forall i \in \mathbb{N}, (3^k \leq i \leq n-3^k \Rightarrow cpos(x+i, 3^k) = red)))$.

補題 4.10 は最上段のマスの色を関数 $colorBYB$ で塗ると，最上段より 3^k 下の段のマスは外側から $n-2 \cdot 3^k+1$ マスはすべて赤で塗られていることを

表している．

証明．$3^k \cdot 2+1 \leq n < 3^{k+1}$ を満たす n をとる．

- $\forall i \in \mathbb{N}, (0 \leq i \leq n-3^k \cdot 2 \Rightarrow cpos(x+i, 3^k) = red)$ を示す．
  $0 \leq i \leq n-3^k \cdot 2$ を満たすように任意に i をとると，$0 \leq i \leq n$，$0 \leq i \leq 3^k-1$ を満たすので，仮定より $cpos(x+i,0) = colorBYB(x,n,k,x+i)$ であり，$colorBYB(x,n,k,x+i) = blu$ が導ける．よって，$cpos(x+i,0) = blu$. また，$0 \leq i+3^k \leq n$，$3^k \leq i+3^k \leq n-3^k$ を満たすので，$cpos(x+i+3^k,0) = colorBYB(x,n,k,x+i+3^k)$ であり，$colorBYB(x,n,k,x+i+3^k) = yel$ が導ける．よって，$cpos(x+i+3^k,0) = yel$. さらに，仮定より $TriangleF(cpos, x+i, 0, 3^k)$ だから $cpos(x+i, 3^k) = mix(cpos(x+i,0) = colorBYB(x,n,k,x+i), cpos(x+i+3^k,0)) = mix(blu, yel) = red$ が成立する．よって，$cpos(x+i, 3^k) = red$.
- $\forall i \in \mathbb{N}, (3^k \leq i \leq n-3^k \Rightarrow cpos(x+i,3^k) = red))$ を示す．
  $3^k \leq i \leq n-3^k$ を満たすように任意に i をとると，$0 \leq i \leq n$，$3^k \leq i \leq n-3^k$ を満たすので，仮定より $cpos(x+i,0) = colorBYB(x,n,k,x+i)$ であり，$colorBYB(x,n,k,x+i) = yel$ が導ける．よって，$cpos(x+i,0) = yel$. また，$0 \leq i+3^k \leq n$，$3^k \leq i+3^k \leq n-3^k$ を満たすので，$cpos(x+i+3^k,0) = colorBYB(x,n,k,x+i+3^k)$ であり，$colorBYB(x,n,k,x+i+3^k) = blu$ が導ける．よって，$cpos(x+i+3^k,0) = blu$. さらに，仮定より $TriangleF(cpos, x+i, 0, 3^k)$ だから $cpos(x+i,3^k) = mix(cpos(x+i,0), cpos(x+i+3^k,0)) = mix(yel, blu) = red$ が成立する．よって，$cpos(x+i,3^k) = red$.

□

補題 4.11 (*LongOddB*)**.** $\forall cpos, \forall x,n,k \in \mathbb{N}, (3^k \cdot 2+1 \leq n < 3^{k+1}) \Rightarrow Fmix(cpos) \Rightarrow (\forall x_1, y_1 \in \mathbb{N}, TriangleF(cpos, x_1, y_1, 3^k)) \Rightarrow (\forall i \in \mathbb{N}, (0 \leq i \leq n \Rightarrow cpos(x+i,0) = colorBYB(x,n,x+i))) \Rightarrow \forall i \in \mathbb{N}, (0 \leq i \leq n-3^k \cdot 2 \Rightarrow cpos(x+i, 3^k \cdot 2) = red)$.

補題 4.11 は最上段のマスの色を関数 $colorBYB$ で塗ると，最上段から $3^k \cdot 2$ 下の段のマスはすべて赤で塗られていることを表している．

証明．$0 \leq i \leq n-3^k \cdot 2$ を満たす i を任意にとると，補題 4.10 より $cpos(x+i,3^k) = red$. また，$3^k \leq i+3^k \leq n-3^k$ でもあるから補題 4.10 より $cpos(x+i+3^k, 3^k) = red$. さらに，仮定より $TriangleF(cpos, x+i, 3^k, 3^k)$ だから $cpos(x+i, 3^k \cdot 2) = mix(cpos(x+i,3^k), cpos(x+i+3^k,3^k)) = mix(red, red) = red$. よって，$cpos(x+i, 3^k \cdot 2) = red$. □

補題 4.12 (*LongOddC*)**.** $\forall cpos, \forall x,n,k \in \mathbb{N}, (3^k \cdot 2+1 \leq n < 3^{k+1}) \Rightarrow Fmix(cpos) \Rightarrow (\forall x_1, y_1 \in \mathbb{N}, TriangleF(cpos, x_1, y_1, 3^k)) \Rightarrow (\forall i \in \mathbb{N}, (0 \leq i \leq n \Rightarrow cpos(x+i,0) = colorBYB(x,n,x+i))) \Rightarrow (cpos(x,n) = red)$.

補題 4.12 は最上段のマスの色を関数 $colorBYB$ で塗ると，最下段のマスは赤になることを表している．

証明．補題 4.11 より $\forall i \in \mathbb{N}, (0 \leq i \leq n-3^k \cdot 2 \Rightarrow cpos(x+i, 3^k \cdot 2) = red)$. さらに，補題 3.12 より $cpos(x,n) = red$. □

補題 4.13 (*LongOdd*)**.** $\forall x,n,k \in \mathbb{N}, (3^k \cdot 2+1 \leq n < 3^{k+1} \wedge odd(n) = true) \Rightarrow \neg WellColoredTriangleF(x,n)$.

証明．補題 3.13 より $\exists cposBYB, Fmix(cposBYB) \wedge \forall x_1, y_1 \in \mathbb{N}, cposBYB(x_1,y_1) = liftpaint(colorBYB(x,n), x_1, y_1)$. さらに，存在する $cposBYB$ をそのまま $cposBYB$ として名付けると，$\forall i \in \mathbb{N}, cposBYB(x+i,0) = colorBYB(x,n,k,x+i)$ を満たす．これより $cposBYB(x,0) = colorBYB(x,n,k,x)$，$cposBYB(x+n,0) = colorBYB(x,n,k,x+n)$. また，$0 \leq 0 \leq 3^k-1$，$n-3^k+1 \leq n \leq n$ より $colorBYB(x,n,k,x) = colorBYB(x,n,k,x+n) = blu$. さらに，仮定より $TriangleF(cposBYB, x, 0, n)$ であるから $cposYBBY(x,n) = mix(cposBYB(x,0), cposBYB(x+n,0)) = mix(blu, blu) = blu$ となる．一方で，定理

4.1，補題 4.12 より $cpos(x, n) = red$ となるので矛盾する．□

以上より補題 4.5，4.9，4.13 から 4.2 節の冒頭で述べた定理 4.2 を証明された．さらに，定理 4.1（十分条件），定理 4.2（必要条件）が成立するので定理 2.5（必要十分条件）も示された．

5 形式化するにあたって

今回の三角形三色問題の形式的証明を完成させるために，問題の形式化や証明の見直しと改訂を何度かくり返すことで，最終的には手書きの証明に近い形のものに到達することができたように思える．本節ではそこに至るまでに遭遇した問題点とその際のいくつかの工夫点，得られた知見について述べる．

5.1 遭遇した問題と工夫点

遭遇した問題とそれに対するための工夫についてまとめると以下の 3 点が挙げられる．以下，逆三角形の左上のマスのことを，その逆三角形の「基準マス」とおよび，その位置を「基準座標」と呼ぶことにする．

1. 十分条件の証明中の数学的帰納法が上手く回らない問題を解決するために述語 *WellColoredTriangleF* と *TriangleF* に基準座標に関するパラメータを追加した．
2. 冗長なコード削減のために最上段の色塗り関数 (*colorYB* や *colorYBBY*) を黄色と青の対称性を利用して簡略化した．
3. 逆三角形の彩色の表現に関して彩色関数を用いて等式変形による証明を基本とすることで，SSReflect が提供する効率的な証明の恩恵を受けられるようにした．

1 について述べる．最終的な定理 (定理 2.5) は n 段の逆三角形の端点の座標が $(0,0),(n,0),(0,n)$ であることから，素朴に形式化しようとすると以下のような段数 n に関する論理式が考えられる：

```
∀ n : ℕ, n>0 →
(∃ k : ℕ,n=3^k) ↔ (WellColoredTriangle' n).
```

ただし，*WellColoredTriangle'* の定義は以下の通り：

```
Definition Triangle' cpos n : bool :=
  (cpos 0 n) = mix (cpos 0 0) (cpos n 0).
Definition WellColoredTriangle' n :=
  ∀ cpos: ℕ → ℕ → Color,
  F_mix cpos → Triangle' cpos n.
```

しかし，この定義だと逆三角形の位置が固定されてしまうため，十分条件の証明においていくつかの部分逆三角形に対して帰納法の仮定を適用する証明を上手く書くことができない．これに対し，逆三角形の位置を自由に動かせるように端点の座標を $(x, y), (x+n, y), (x, y+n)$ として *TriangleF* には x, y をパラメータとして追加，*WellColoredTriangleF* には x をパラメータとして追加することで解決した．

2 について述べる．最上段の色塗り関数 *colorYB* や *colorYBBY* による色塗りは黄色と青を逆転させても議論の本質は変わらず，対称的である．この性質を利用して関数の定義を大幅に簡略化をおこなった．関数 *colorYB*(最新版では coloringYB と変更) の定義は

```
Definition coloringYB n x :=
  if (x ≤ n) && ~~ odd x then yel else blu.
```

で与えた．この定義は基準座標の x 座標が偶数なら「黄青黄…」と塗り，奇数なら「青黄青…」と塗ることを想定している．前述の 1 より逆三角形の基準座標は $(0, 0)$ とは限らないため，任意の場所を基準座標に取っても議論が成立することが必要になる．簡略化する前の関数 *colorYB* では黄色から必ず開始すると固定していたため，対象座標の基準座標からの相対位置を得るために x 座標の差分を計算し，その差分の偶奇で場合分けをしていた．一般に自然数の差分の計算は複雑化の要因となるが，この点が解消されたおかげで簡略化に成功した．

3 について述べる．一般的に，形式化を行う際の方針として論理式を用いる方法と関数を用いる方法がある．三角形三色問題の場合は「座標 (x, y) のマスの色は c だ」を意味する彩色述語 $Cpos(x, y, c)$ を導入することも考えられたが，その場合は「各マスには必ず色が塗られる」や「1 つのマスに 2 色以上は塗られない」など，この述語が満たすべき彩色条件を常に仮定する必要があり，冗長なコードの要因となる †6．

†6 我々の初期の実装ではこの方針を採用していてかなり

今回の問題では彩色関数を用いることで，(1) 関数により彩色の状況が簡潔に表現でき，かつ彩色条件の記述が不要，(2) 論理式の変形ではなく等式変形を基礎とした証明になるため，SSReflect が提供する効率的な証明の恩恵を受けることができる．この 2 点の恩恵により不要な議論を減少させることができ，スムーズな証明を得ることができた．

5.2 Coq で形式的証明を行うことの恩恵

手書きの証明と Coq による形式的証明を比較して，形式的な証明を行うことの恩恵について，これまでに得られた知見を述べる．既に述べた 81 通りの場合分けの処理や SSReflect の reflection 機能以外の恩恵に関しては以下の点が挙げられる．

1. 通常の数学の証明と異なり，形式的証明においては主張の正しさの先の部分を気にすることができ，形式的証明についてコードレビューを通した「形式化はスムーズか」や「証明は効率的か」などの議論をしたり，定理の共有ができる．
2. SSReflect が提供するタクティクとタクティカルを用いて複数の細かい操作を一度に行うことで形式的証明の厳密性を保ちながら紙の証明と近い感覚で議論を進めることができる．

1 について述べる．通常の数学の証明は定理が正しいかどうかに主眼が置かれる一方で，形式的証明においては正しさは定理証明支援系により保証されることから正しさの先の部分を気にすることができる．上手い形式化を採用することで定義や補題が簡潔に記述できているか，タクティクを有効に使って効率よく証明を行えているかなどが議論できる．このような議論の場を得ることができる点が定理証明支援系を用いることの恩恵の 1 つである [†7]．また，形式的に証明された定理はライブラリとして集積・共有することで後の人々に貢献することができる．

2 について述べる．紙の上の証明では場合分けの単純な場合を「明らか」として済ませたり，暗算できるような自明な等式であればわざわざ書くことなしに用いてゴールを書き換えるなどの一足飛びをすることは普通であるが，Coq で証明を行う場合はこのような小さく飛ばすステップも明示的に記述する必要がある．SSReflect が提供するタクティクを用いることで複数の細かい操作を一度に行うことができ，厳密でいながら紙の証明と近い感覚で議論を進めることができる．本研究においてコードの改善をする中でこの機能を積極的に用いることで，当初とは見違えるほどの効率化をすることができた．機能を利用した具体例を挙げると以下の通りである．

- いくつかのサブゴールに分割し，その後のいくつかの処理も同時にするときはタクティカル [|] を使って短く記述した．典型的な例としては以下が挙げられる：

```
Let longodd_red_both_sides :
(∀ i, i ≤ n − (3 ^ k).*2 →
colfun (x + i) (3 ^ k) = red) ∧
(∀ i, 3 ^ k ≤ i ≤ n − 3^ k →
colfun (x + i) (3 ^ k) = red).
Proof.
split⇒
[i i_range_right| i /andP
  [i_range_left i_range_right]];
…
Qed.
```

  これは「ゴールを split で 2 つのサブゴールに分割し，1 つ目のサブゴールに move=> i i_range_right を適用する，2 つ目のサブゴールには move=> i. move=> /andP [i_range_left i_range_right]. を適用する」を同時に行っている．紙の上の証明でもこれくらいの処理を明示せずにやると思われるが，この処理を厳密性を確保しつつ同様の感覚で簡潔に書くことができるようになる．

- タクティク have とタクティカル -> (もしくは <-) の組合せで have の内容を示した後に自動的に示した内容で等式を変形する機能を積極的に利用した．特に have の内容が明らかな場合はタクティカル // を利用することで瞬時に処理することができて重宝した．使用例としては以下が挙げられる：

冗長であった．

†7 実際，今回の内容は査読者の方々によるコードレビューにより SSReflect を効率的に使ったコードに改善することができた．この場をお借りして感謝したい．

```
Lemma TCTP_nec_longodd x n k :
(3 ^ k).*2.+1 ≤ n < 3^ k.+1 →
~ WellColoredTriangle x n.
Proof.
...
have → : colfun x n = red;
first by exact: (longodd_bottom _ k).
have → : colfun x 0= blu
by rewrite −(addn0 x)
−topcolor// BYB_blu_left.
have → // : colfun (x +n) 0= blu.
...
Qed.
```

1 つ目の `have` は主張 `colfun x n = red` を導入した直後に証明し，ゴールをこの等式を用いて変形している．2 つ目も同様に行っている．さらに 3 つ目は `//` により自明な主張の証明を瞬時に済ませている．これくらい簡潔に記述できることは，完成後の証明の短縮化につながることはもちろんのこと，多数の細かく (当たり前に見える) 処理に追われることに対するユーザのストレス軽減にも繋がることを実感した．

6　まとめ

本研究では *cpos* や *liftpaint* といった関数等を適切に定めることで三角形三色問題の幾何的な状況を再現しつつ，三角形三色問題の証明の形式化を Coq + SSReflect を用いて完成させることができた．

謝辞

本論文は日本ソフトウェア科学会第 38 回大会 (JSSST2021) の質疑応答および査読者の方からいただいたコメントを基に大会発表論文から見直しと改訂を行いました．貴重なコメントをいただけたことに感謝いたします．

参 考 文 献

[1] “The Coq Proof Assistant”, https://coq.inria.fr/.
[2] “The SSReflect proof language”, https://coq.inria.fr/refman/proof-engine/ssreflect-proof-language.html.
[3] 萩原学, アフェルド・レナルド, “Coq/SSReflect/-MathComp による定理証明”, 森北出版, 2018.
[4] 西山豊, “エレガントな解答をもとむ 出題 2”，数学セミナー, 4 月号, pp.87–91, 2013.
[5] 西山豊, “数学を楽しむ/三角形三色問題”, 現代数学, Vol.47, No.10, pp.36–41, 2014.
[6] Y. Nishiyama, “The Three-Color Triangle Problem”, International Journal of Pure and Applied Mathematics, Vol.85, No.1, pp.69–81, 2013.