

gyoudori パッケージ

行取り（行ドリ）の実現

木枝祐介

2026 年 10 月 26 日 v20261026.0

概要

gyoudori は見出し・ディスプレイ数式・フロートを行取り（行ドリ）で配置し、その前後の本文の行を基本版面の行位置からずらさないための L^AT_EX パッケージである。pLaTeX, upLaTeX, LuaLaTeX (LuaTeX-ja) で動く。このパッケージは横組用である。英語版のマニュアル (gyoudori-en.pdf) もある。

1 行取りとは何か

行取り（行ドリ）とは恣意的なレイアウトのひとつで、見出しのように本文と大きさの違う要素が行送り方向に占める領域を基本版面の何行分かとして決め、その後ろの本文の行を基本版面で決めた行の位置（行のグリッド）からずらさないようにする方法である。

行取りは日本語の書籍などの組版で、各ページの本文の行を行のグリッドにそろえて配置するように指定されたときに用いられる。このパッケージはそうした指定のもとで組版しなければならない場合のための道具である。

このパッケージは横組用である。縦組は考慮していない（縦組で試した結果は 8 節に記す）。

1.1 行取りの領域

n 行取りの領域の行送り方向の大きさは JLREQ (W3C『日本語組版処理の要件』) §4.1.6 の定義のとおり

$$\text{字の枠の高さ} \times n + \text{行間} \times (n - 1) = \text{行送り} \times (n - 1) + \text{字の枠の高さ}$$

である。つまり、 n 行分の行の位置のうち、1 行目の字の枠の上端から n 行目の字の枠の下端までの範囲である (図 1)。見た目にはページ（段）の途中ではこの領域の前後に行間が加わり、ページ（段）の先頭では後ろにだけ行間が加わる。図の行送りはベースライン（破線）どうしの間隔で示している。このマニュアルでは行送りを測る基準点を仮にベースラインとしている (1.3 節)。



図1 3行取りの領域と、その中央に置いた見出し（3行取り中央）

このパッケージは既定では内容を領域の中央に置く．位置の指定（t, c, b）で，上寄せと下寄せに変えられる（図2）．領域の中のどこに置くかは組版の指定によって異なり，JLREQ も中央のほかの配置方法をあげている（3.3 節）．

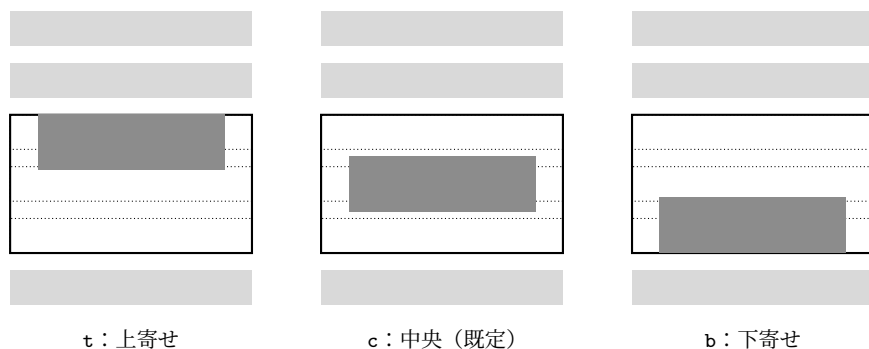


図2 領域の中での位置

1.2 このパッケージでの寸法

行送りには `\normalsize` の `\normalbaselineskip` を，字の枠の高さには `\normalsize` の「あ」の高さ + 深さ（JFM の寸法）を使う．文中で `\small` などに切り替えていても基準は本文の行のグリッドである（ディスプレイ数式と，`gyoudori` 環境・`\gyoudoriarea` に `local` をつけた場合を除く．3.1 節，4.3 節）．

字の枠の高さは JFM によって異なる．`upLaTeX`，`LuaTeX-j`（`jlreq` を含む）の標準の JFM では全角幅と同じ（正方形の字の枠）で，JLREQ の定義と一致する．`pLaTeX` の標準の JFM（`jis`）では全角幅の 0.952 倍である．このパッケージは JFM の字の枠に従うので，`pLaTeX` では領域が JLREQ の定義より全角幅の 0.048 倍だけ短くなる．

1.3 行送りの基準点と `\baselineskip`

JLREQ (§2.4.2 の注) は行送りを「隣接する行同士の基準点から基準点までの距離」としているが、基準点そのものは定めていない。基準点は処理系により異なるとしたうえで、横組では字の枠の天地中央とする例が多いと述べているが、その根拠は示されていない。また、 $\text{\LaTeX}$ ($\text{\TeX}$) は行をベースラインで並べるので、天地中央を基準点とする考え方とは大きく異なる。

そこでこのマニュアルでは行送りの基準点を仮にベースラインとする。 $\text{\LaTeX}$ の既定の動作もこのパッケージの実装も、ベースラインを基準にしているからである。つまり、行送りはベースラインどうしの間隔である。

$\text{\LaTeX}$ の `\baselineskip` は隣接する行のベースラインの間隔である。 $\text{\TeX}$ は前の行の深さと次の行の高さから、ベースラインの間隔が `\baselineskip` になるように行の間にグルーを入れる。ただし、行の間が `\lineskiplimit` より狭くなるときは代わりに `\lineskip` を入れるので、間隔は `\baselineskip` より広がる。

このパッケージは本文の行送り b として `\normalsize` の `\normalbaselineskip` を使い (`\linespread` で変えた場合も反映される)、ベースラインが b の間隔で並ぶ位置を本文の行の位置 (行のグリッド) とみなす。各行の字の枠はベースラインから上に「あ」の高さ h 、下に深さ d の範囲である。したがって n 行取りの領域は 1 行目のベースラインの h 上から n 行目のベースラインの d 下までで、大きさは $(n-1)b + h + d$ である。領域は 1 行目のベースラインが直前の行のベースラインから b の位置に来るように置き、領域の後の行のベースラインが n 行目のベースラインから b の位置に来るようにする (7.2 節)。ディスプレイ数式も式の 1 行目と最後の行のベースラインの中点を、領域の 1 行目と最後の行のベースラインの中点に合わせて置く (7.3 節)。

同じ大きさの文字だけが並ぶ行どうしでは基準点をどこにとっても行送りの値は変わらない。JLREQ の関係 (行送り = 文字サイズ + 行間) も領域の大きさの式も、基準点によらない。基準点の違いが効くのは大きな見出しのように文字の大きさが違う行を含むときで、ベースラインの間隔と天地中央の間隔は一致しなくなる。このパッケージはそうした見出しなどを行取りの領域に収め、前後の本文の行をベースラインで行のグリッドにそろえる。

また、このパッケージの行間 (行送り - 字の枠の高さ) は字の枠が正方形の JFM では JLREQ の行間 (行送り - 文字サイズ) と一致する。 pLaTeX の標準の JFM では字の枠の高さが全角幅の 0.952 倍なので、JLREQ の行間より全角幅の 0.048 倍だけ大きくなる。

2 動作環境とインストール

- pLaTeX , upLaTeX , または $\text{LuaLaTeX} + \text{LuaTeX-j}\text{\LaTeX}$ (`l\text{\LaTeX}article`, `jlreq` など)。横組のみを対象とする。
- $\text{\LaTeX}$ 2022-06-01 以降 (パッケージオプションにカーネルの key-value 機能 `\DeclareKeys`, `\ProcessKeyOptions` を、環境の前後の処理に環境フック `\AddToHook` を使う)。ほかのパッケージは必要ない。

gyoudori.sty を TEXMF ツリー（例えば tex/latex/gyoudori/）に置く．

3 使い方

3.1 gyoudori 環境

```
\begin{gyoudori}[<オプション>]{<行数>}
```

<内容>

```
\end{gyoudori}
```

<内容> を <行数> 行取りする．<行数> は 1 以上の整数で，そうでなければエラーになる．<オプション> には次のものをカンマで区切って指定する．

位置 t（上寄せ），c（中央．既定），b（下寄せ）のいずれか．

local 基準をその場の文字サイズにする（下記）．

indent, noindent <内容> の段落を字下げするかしないか．既定はパッケージオプション indent による（4.2 節．既定は字下げしない）．

<内容> は垂直モードで組まれるので，複数行や文字サイズの変更を含められる．<内容> は字の枠で位置を決める．1 行目のベースラインの上に <内容> の最後の文字サイズの「あ」の高さ，最後の行のベースラインの下に「あ」の深さをとった範囲を，領域の中で位置の指定に従って置く．このため，欧文だけの見出しも和文の見出しと同じ位置に置かれる．<内容> の字面が領域の外側の行間に半分を超えてはみ出すときは警告を出す（パッケージオプション overflow でエラーにできる．4.1 節）．

local をつけると，\normalsize の行送りではなく，その場の文字サイズの「あ」の高さ・深さと \baselineskip を基準に行取りする．\small などに切り替えた部分のなかで，まわりの行にそろえるときに使う（例：\begin{gyoudori}[b,local]{2}）．local をつけないときは文字サイズによらず \normalsize の行送りが基準である．

本文の行である．前の段落の最後の行である．

3 行取り中央の見出し

本文の行である．見出しの後の段落も行の位置からずれない．

2 行取り下寄せの見出し

空行を置かずに続けた本文は字下げされない．

次は \small で組んだ部分のなかで local をつけた例である．グレーの帯で \small の行のグリッドを示す．

`\small` で組んだ本文の行である.

3 行取り中央の見出し

続く本文の行も `\small` のグリッドにのる.

- `\end{gyoudori}` の直後に空行がなければ, 後続の文は字下げせず `\parskip` も入れない (ディスプレイ数式の後と同じ扱い). 空行があれば, 新しい段落として字下げし `\parskip` も入れる.
- 領域は分割されない. ページ (段) の末尾に領域が入りきらなければ, 見出し全体が次のページ (段) の先頭に送られる (JLREQ §4.1.7 の a, c).
- 見出しの直後では改ページ・改段しない. 見出しの後ろに本文の行が 1 行も入らないときは見出し全体が次のページ (段) の先頭に送られる (JLREQ §4.1.7 の b). いずれの場合も前のページ (段) の末尾は空いたままになる.

3.2 `\gyoudoriarea` 命令

`\gyoudoriarea`[<オプション>]{<行数>}{<内容>}

`gyoudori` 環境と同じく行取りする. <オプション> も同じである. 命令なので, 直後の字下げの制御はない.

3.3 JLREQ の配置方法との対応

JLREQ §4.1.6 は行取りによる別行見出しの配置方法として a から d をあげている. このパッケージでは次のように書ける.

- 複数行の領域の中央** `\begin{gyoudori}{3}` (3 行取り中央).
- 中央に置き, 前後に行単位の空きをとる** 行送りの整数倍の空きを組みあわせる. 例えば前を 1 行アキにするなら, `\vspace{\baselineskip}` の後に `gyoudori` 環境を置く.
- 領域の中の指定位置** 上寄せ (t) と下寄せ (b) を用意している. それ以外の位置は <内容> の中に空きを入れて調整する.
- 1 行の見出しを行の位置に置き, 前を 1 行アキ** 本文と同じ大きさの見出しを 2 行取り下寄せ (位置 b, 行数 2) にすると同じ配置になる.

見出しとしての行取りがページの先頭に来たとき 見出しの行取りの領域もページ (段) の先頭では版面の天につき, 領域の中の位置は位置の指定に従う (3.4 節). 例えば d の「前を 1 行アキにした小見出し」(2 行取り下寄せ) がページの先頭に来ると, 1 行目が空き見出しは 2 行目に来る. これは JLREQ §4.1.8 の「常に 1 行アキにする」方針に対応する.

3.4 ページの先頭・末尾での行取りの領域

見出しに限らず, `gyoudori` 環境, `\gyoudoriarea`, ディスプレイ数式の行取りの領域はページ送りの都合で偶然ページ (段) の先頭や末尾に来ることがある. そのときの挙動は次のとおりである. フロートの置き場所は $\text{\LaTeX}$ が決めるのでここでは扱わない (4.5 節).

行取りの領域がページ先頭に来たとき 領域そのものが版面の天につき (天ツキ), 領域の前には行間が入らない. 領域の中の位置はページの途中と同じく位置の指定 (`t`, `c`, `b`) に従う (図 3). n 行取りの次の行は $n + 1$ 行目から始まる. ディスプレイ数式も同じで, 数式は領域の中でページの途中と同じ位置に置かれる.

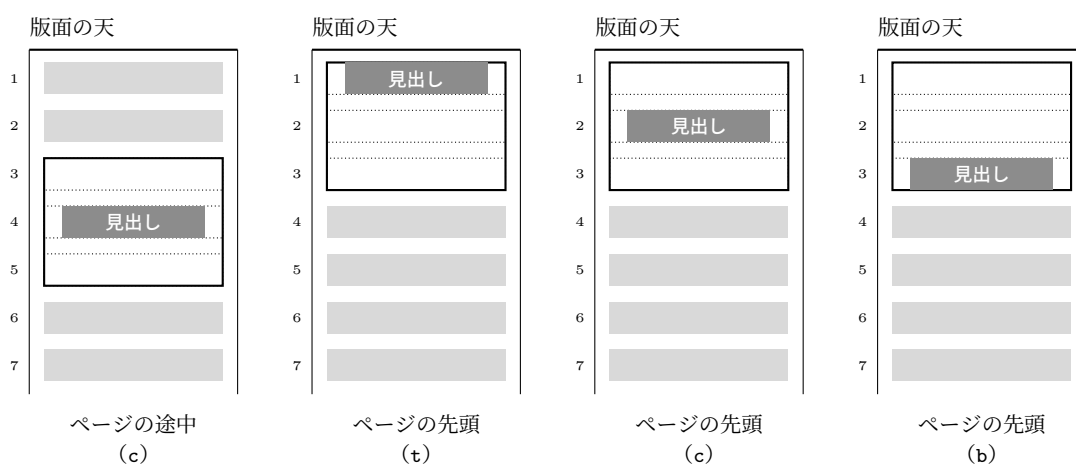


図 3 3 行取りの領域がページの途中と先頭に来たとき (左の数字は行番号)

行取りの領域がページ末尾に来たとき 領域は分割しない. 領域が版面の最終行までに収まればそのページに置き, 収まらなければ領域全体を次のページ (段) の先頭に送る. このとき前のページ (段) の末尾は入らなかった行数だけ空いたままになる.

- `gyoudori` 環境と `\gyoudoriarea` は見出しとして使うことを想定して直後では改ページしない. 領域が最終行で終わり, 後ろに本文の行が 1 行も入らないときも領域全体を次のページ (段) に送る (3.1 節).
- ディスプレイ数式の後では改ページしてよい. 領域が最終行で終わればそのページに収まり, 続く本文は次のページ (段) の 1 行目から始まる. 複数行の数式も 1 つの領域なので, `\allowdisplaybreaks` を指定しても途中で改ページしない.

4 パッケージオプション

オプションは key-value 形式で指定する. 真偽値のオプションはキーだけを書くと `true` になる.

```
\usepackage[displaymath,float,textfloatsep=2\baselineskip]{gyoudori}
```

4.1 overflow=warning|error

gyoudori 環境・\gyoudoriarea の〈内容〉が領域に収まらないとき（字面が領域の外側の行間に半分を超えてはみ出すとき）に、警告を出すか（warning. 既定）、エラーにするか（error）を指定する。行数は自動では増やさない。

4.2 indent=true|false

gyoudori 環境・\gyoudoriarea の〈内容〉の段落を字下げするかを指定する。既定は false（字下げしない）である。環境ごとに indent, noindent で変えられる（3.1 節）。

4.3 displaymath=true|false

ディスプレイ数式を自動的に行取りする。既定は false である。

対象 equation, eqnarray, \[と \] で囲む数式（displaymath 環境）と、amsmath の equation*, align, gather, multiline, flalign, alignat, およびそれぞれの * つきである。\$\$ で囲む数式は対象外である。式番号, \label, \eqref は、このオプションを使わないときと同じように使える。

行数 数式が要する行数に 1 行を足した行数で行取りする。2 行に満たなければ 2 行とする。足す行数は displaymathextra で変えられる（4.4 節）。数式が要する行数とは、その行数の領域の外側の行間に半分まではみ出して数式が収まる最少の行数である。本文の行の字面が字の枠からはみ出すのと同じ扱いである。例えば、1 行で足りる式は 2 行取り、2 行を要する式は 3 行取りになる。式の上下には、合わせて 1 行分のアキが加わる。数式の前後の \abovedisplayskip など使わない。

位置 数式はその 1 行目と最後の行のベースラインの中点を、領域の 1 行目と最後の行のベースラインの中점에合わせて置く。1 行の式は奇数行取りなら式のベースラインが中央の行のベースラインにのり、偶数行取りなら 2 つの行のベースラインの中間に来る。式番号が下の行に付いた式など、上下が非対称でこの位置では収まらない式は収まる範囲に寄せて置く。

測り方 amsmath が各行に入れる支柱（\strut）は「あ」の字の枠の大きさにして測るので、見えない支柱の深さで行数は増えない。式の直前に hyperref のリンクの着地点などが入ってもそれを 1 行と数えることはない。

行取りの基準 その場の文字サイズの「あ」の高さ・深さと \baselineskip である。 \normalsize の本文では \normalsize の行送りと同じになり、脚注や \small で組んだ解答などのなかではまわりの行送りにそろふ。

本文の行である．次の式は行取りされる．

$$a = b + c \quad (1)$$

$$d = e \quad (2)$$

式後の本文もグリッドにのる．

$$x = y$$

本文の行である．

グレーの帯で行のグリッド（各行の字の枠）を示す．この例では 2 行の `align` は 3 行取り，1 行の `gather*` は 2 行取りの領域の中央に置かれ，その前後の本文はどれもグリッドにのっている．

分数や積分，総和を含む大きな数式も要する行数に 1 行を足した行数で行取りされる．

本文の行である．積分と総和の式である．

$$\int_0^1 f(x) dx = \sum_{k=1}^n \frac{a_k}{k!} \quad (3)$$

式後の本文である．次は 2 行の式である．

$$S_n = \sum_{k=1}^n k = \frac{n(n+1)}{2} \quad (4)$$

$$I = \int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi} \quad (5)$$

続く本文である．場合分けの式である．

$$f(x) = \begin{cases} \frac{x^2}{2} & (x \geq 0) \\ -x & (x < 0) \end{cases} \quad (6)$$

最後の本文の行である．

この例では上から 3 行取り，5 行取り，3 行取りになっている．どの場合も次の本文はグリッドにのる．

次は `\small` で組んだ部分のなかの例である．グレーの帯で `\small` の行のグリッドを示す．

`\small` で組んだ解答の行である．

$$\int_0^1 x^2 dx = \frac{1}{3}$$

続く行も `\small` のグリッドにのる．

$$x = y$$

最後の行である．

4.4 `displaymathextra=<整数>`

`displaymath` で、ディスプレイ数式が要する行数に足す行数を指定する (0 以上の整数)。既定は 1 である。行取りの行数は「要する行数 + この値」で、2 行に満たなければ 2 行とする。0 にすると、数式が収まる最少の行数 (2 行以上) になり、式と前後の行の字の枠との間は行間の半分まで詰まることがある。2 以上にすると、その分だけ式の上下のアキが広がる。

```
\usepackage[displaymath,displaymathextra=0]{gyoudori}
```

4.5 `float=true|false`

フロートを行取りで配置する。既定は `false` である。以下ではページ (段) の上に置かれたフロートを「上のフロート」、下に置かれたものを「下のフロート」、本文の途中 (配置の指定 `h` の位置) に置かれたものを「本文中のフロート」と呼ぶ。

- フロートはそれが収まる最少の行数 (1 行以上) の行取りの領域の大きさにする。
- 置き場所が決まったら、領域の中での位置を次のようにする。上のフロートは上寄せで、領域の上端を本文 1 行目の字の枠の上端にそろえる。下のフロートは下寄せで、領域の下端を本文の行の字の枠の下端にそろえる。本文中のフロートは中央で、前後の本文の行にそろえる。
- フロートまわりの 5 つのアキ (`\floatsep` など。次節) は「行間 + 行送りの整数倍」のうち指定値以上で最小の値に丸め、伸縮をなくす。`\intextsep` はフロートの上下に入るので、行間を半分ずつ割り振った値になる。
- フロートだけのページ (段) ではフロートを上寄せにし、最初のフロートの領域の上端を本文 1 行目の字の枠の上端にそろえる。フロートの間のアキ (`\@fpsep`) も同じように丸める。
- 版面より高いフロートは $\text{\LaTeX}$ と同じく警告を出して縮めて置く。その上端を行の位置にそろえるため、 $\text{\LaTeX}$ より少し (7.4 節の `\@fptop` の分) 多く縮める。

これにより、フロートの前後で本文の行は行の位置からずれない。 $\text{\LaTeX}$ は `[h]` だけの指定を `[ht]` に変える。その場に置けずに上に回ったフロートは上寄せで組み直される。本文中のフロートの直前では改段しない。

4.6 `floatsep, textfloatsep, intextsep, dblfloatsep, dbltextfloatsep`

同名の長さを設定する。指定しなければ、クラスファイルの値を使う。`float=true` のときはこの値をもとに上記の丸めを行う。

5 ほかのパッケージとの併用

以下は `float=true` のときの動作である。

nidanfloat 段抜きのフロート (`figure*`) は上なら版面の上端に、下なら版面に入る最終行の字の

枠の下端にそろえる。2 段組の段ごとの下のフロートは段に入る最終行の字の枠の下端にそろえる。読み込み順は問わない。

stfloats 段抜きのフロートの上と下を同じようにそろえる。

flushend 最終ページでそろえた段の中のグルーを伸縮させず、行を行の位置に保つ。spread, shrink 系のオプションは効かない。左右の段のつなぎ目の間隔は行送りの整数倍に切り上げる。

evenend (LuaLaTeX) 最終ページでそろえた段のフロートを、上なら上寄せ、下なら下寄せにそろえる。段数の切り換えや最終ページで持ち越されて段の上に置かれる段内フロートも上寄せにそろえる。

6 JLREQ との関係

JLREQ の行取りに関する記述と、このパッケージの動作を照合した。

行送りの基準点 (§2.4.2 の注) JLREQ は基準点を定めていない。このパッケージはベースラインを基準点とする (1.3 節)。

領域の大きさ (§2.5.2, §4.1.6) 一致する。字の枠には JFM の寸法を使うので、pLaTeX の標準の JFM では正方形の字の枠を前提にした定義より全角幅の 0.048 倍だけ短くなる (1.2 節)。

領域の中の配置 (§4.1.6 の a から d) a は既定の中央で、c は上寄せ・下寄せで対応する。b と d は組みあわせで実現する (3.3 節)。

ページ末・段末の見出し (§4.1.7) 一致する。領域が入りきらないとき、見出しの後ろに本文の行が 1 行も入らないときは見出し全体を次のページ (段) の先頭に送る。縦組の偶数ページの例外は横組のみが対象なので扱わない。

ページ先頭の見出し (§4.1.6, §4.1.8) 一致する。領域は版面の天につき、その後ろにだけ行間が加わる (§4.1.6)。前を 1 行アキにした小見出しでは §4.1.8 の 3 つの方針のうち「常に 1 行アキにする」に対応する (3.3 節)。

段の末尾の空き (§4.1.7 の c) JLREQ は段末を空けないための何らかの処理を望ましいとしつつ、やむを得ない場合は空けてよいとしている。このパッケージは空けたままにする。

別行の数式 (§4.1.6 の Note) JLREQ は文字サイズより大きい別行の数式について、行取りで指定する方法と、前後に最低の行間をとって半端を版面の中で調整する方法をあげている。displaymath は前者である。数式が要する行数に 1 行を足すこと (displaymathextra で変えられる) と、最少 2 行とすることはこのパッケージの設計である。

図版 (§2.5.2, §4.3) JLREQ は図版などの行送り方向の占める領域も基本版面の行の位置にそろえるべきだとしている。float はこれに従う。図版の前後に本文が 1 行だけ来る配置を避けること (§4.3) は L^AT_EX のフロート配置に任せており、扱わない。

7 仕組み

7.1 記号

b : 行送り (`\normalsize` の `\normalbaselineskip`), h, d : `\normalsize` の「あ」の高さと深さ (字の枠), $g = b - h - d$: 行間, $A(n) = (n - 1)b + h + d$: n 行取りの領域の高さ, t_1 : 版面 (段) の上端から本文 1 行目のベースラインまで (`\topskip` と h の大きいほう. $\text{T}_{\text{E}}\text{X}$ は 1 行目の高さが `\topskip` より大きいと, ベースラインを行の高さの位置に置く).

特定のクラスの設定は前提にしていない. クラスについて前提にしているのは次のことだけである.

- 本文の行送りが, `\normalsize` を実行したときの `\normalbaselineskip` になること (文字サイズを `\fontsize` や `\@setfontsize` で定める通常の定義ならそうなる).
- 本文の行の高さ・深さが「あ」の高さ・深さ (JFM の字の枠) であること.
- 本文の段落の間の `\parskip` が 0 であること (0 でなければ, このパッケージによらず段落ごとに行の位置がずれる).

版面の高さ (`\textheight`) が行送りの整数倍にそろっている必要はない. `\topskip`, `\maxdepth`, フロートまわりのアキはクラスの値をそのまま使う (アキは丸める).

7.2 行取りの領域の置き方

内容を高さ $A(n)$ の `\vbox` に組み, その高さを h に, 深さを 0 につけ替える (中身は下にはみ出して描かれる). 段落を閉じてから, この箱, 行送りの $n - 1$ 倍のカーン, 高さ 0・深さ d の空の箱を続けて置く. $\text{T}_{\text{E}}\text{X}$ は直前の行のベースラインから b の位置に箱のベースライン (領域の 1 行目のベースライン) が来るように `\baselineskip` のグルーを入れる. 空の箱が領域の最後の行にあたるので, 次の行はそのベースラインから b の位置, つまり $n + 1$ 行目に来る. ページ (段) の先頭では `\topskip` によって領域の上端が 1 行目の字の枠の上端につく. 3 つの要素の間では改ページできず, ページ分割の上では n 行の本文の行と同じに扱われる. この置き方は阿部紀行氏のアルゴリズム (<https://abenori.blogspot.com/2017/12/gyodori.html>) を参考にし, 改良を加えたものである. 阿部氏のアルゴリズムは領域を深さの大きい 1 つの箱で表すが, 深さが `\maxdepth` を超える箱はページ分割で最終行の下にはみ出すことがあるので, 3 つの要素に分けた.

`gyodori` 環境と `\gyodoriarea` では `\content` の箱の先頭に `\penalty-10000` を置き, ディスプレイ数式と同じ方法 (7.3 節) で 1 行目と最後の行のベースラインの距離 Δ を求める. `\content` の最後の文字サイズの「あ」の高さ・深さを h_c, d_c とし, 字の枠の範囲 (高さ $\Delta + h_c + d_c$) を位置に応じて領域の上端・中央・下端にそろえる. `local` をつけたときは h, d, b にその場の「あ」の高さ・深さと `\baselineskip` を使う.

`gyodori` 環境と `\gyodoriarea` は置いた直後に `\nobreak` を入れて, 見出しの直後での改ペー

ジを禁じる。gyoudori 環境と数式の環境の後では L^AT_EX の \@doendpe と同じ仕組みで、空行がなければ次の段落を字下げせず、\parskip も 0 にする。

7.3 ディスプレイ数式

h , d , b にはその場の「あ」の高さ・深さと \baselineskip を使う。

1. 環境フック (env/<環境名>/before と after) で、環境の外側に \vbox を開き、その中で \abovedisplayskip など を 0 にして数式を組む。箱の中では \strutbox を「あ」の字の枠の大きさにし (amsmath の各行の支柱)、\predisplaypenalty を -10000 にする。
2. T_EX は式の直前に \predisplaypenalty を入れるので、そこで \vsplit して、式の前の部分 (式の前に段落ができたときの行など) と式の部分に分ける。以下では式の部分だけを測る。式の前の部分は最後に高さ 0 の箱にして式の先頭に重ねて残す。
3. 式の部分の先頭に \penalty-10000 を置いた複製を、\splittopskip を十分大きな値 S にして \vsplit すると、残りの先頭に「 $S-1$ 行目の高さ」のグルーが入る。そこで「残りの高さ $-S$ 」が、数式の 1 行目と最後の行のベースラインの距離 Δ になる。
4. 数式の箱の高さを H 、深さを D とすると、最後の行のベースラインは箱の上端から H 、1 行目は $H-\Delta$ の位置にある。領域の上端から箱の上端までの距離を

$$y = h + \frac{(n-1)b}{2} - H + \frac{\Delta}{2}$$

とすると、数式のベースラインの midpoint が領域のベースラインの midpoint にそろう。

5. $k=1$ から順に増やし、 $H+D \leq A(k)+g$ となる最小の k (数式が要する行数。箱は領域の外側の行間に半分まではみ出してよい) を求め、 $n=k+e$ とする。 e は displaymathextra の値 (既定は 1) で、 $n < 2$ なら $n=2$ とする。 y が $-g/2$ より小さければ $-g/2$ に、 $A(n)+g/2-H-D$ より大きければその値にして、箱を収まる範囲に寄せる。
6. 箱を領域の中で y だけ下げて、7.2 節と同じ方法で置く。

7.4 フロート

大きさ \@endfloatbox の後で、フロートの箱を、高さ + 深さ $\leq A(n)$ となる最小の n (1 以上) の高さ $A(n)$ の \vbox に組み直す (中身は上寄せ)。

組み直し 置き場所が決まったら、箱の外形 (高さ + 深さ) を保ったまま、中身を位置 p (t , c , b) に置き直し、さらに δ だけ下げる。外形を変えないので L^AT_EX のフロート配置の計算は変わらず、段抜きフロートの印 (深さ 1sp) も保たれる。 \@endfloatbox を通らずに積まれた箱 (\twocolumn の段抜きの表題など) は中身を取り出せないで組み直さない。

上のフロート (\@cflt) $p = t$, $\delta = t_1 - h$. 領域の上端が本文 1 行目の字の枠の上端に来る。

下のフロート (\@cflb) $p = b$, $\delta = d - d_{\text{last}}$. d_{last} は段の最後の箱の深さである (段の箱の深さは \maxdepth で打ち切られていることがあるので、最後の箱を取り出して測る)。フロートの

並びの末尾には高さ $0 \cdot$ 深さ d の空の行と、「 $r - d$ 」のグルーを置く。 r は「段の高さ $-t_1$ 」を b で割った余りである。こうすると、`\clearpage` の `\vfil` で押し下げられたときも並びの下端が段に入る最終行の字の枠の下端にそろい、字の枠の下端が版面からはみ出すクラスでも Overfull にならない。押し下げられた場合（段の末尾がグルー）は $\delta = 0$ とし、`\@textbottom` の伸びを打ち消す。

本文中のフロート (`\@addtocurcol`) $p = c$, $\delta = g/2 + d - d_{\text{prev}}$. d_{prev} は直前の行の深さ (`\@pagedp`) である。フロートの直前には `\nobreak` を入れさせ、段の先頭に来ないようにする。

アキの丸め 各アキを、 $g + kb$ (k は 0 以上の整数) のうち指定値以上で最小の値にし、伸縮をなくす。`\intertextsep` はフロートの上下に入るので $g/2 + kb$ とする。こうすると「フロートの領域 + アキ」が行送りの整数倍になり、本文の行がずれない。

フロートだけのページ `\@fptop` を $t_1 - h$ (伸縮なし) にし、`\@fpsep` をほかのアキと同じく丸め、`\@fpbot` に伸びを残す。フロートを積む `\@wtryfc` で中身を上寄せに組み直す（本文中のフロートとして試したときの組み方が残っているため）。収まるかの判定 (`\@xtryfc`) は `\@fptop` を数えないので、判定の間だけ段の高さから差し引く。L^AT_EX は版面より高いフロートを版面の高さに縮める (`\@largefloatcheck`) が、この判定と合わせて `\@fptop` の分も縮め、警告を出す。段抜きフロートだけのページも同じである。

脚注の積み順 pL^AT_EX の `\@makecol` はフロートの後に脚注を積むので、 r を求めるときに脚注の高さを段の高さから引く。L^AT_EX 2025-06-01 以降の `\@makecol` (LuaL^AT_EX など) は脚注の後にフロートを積むので、下のフロートがある段では脚注の前に `\vfil` を入れ、脚注と下のフロートを版面の下端に置く。

7.5 ほかのパッケージとの接続

段抜きフロート (`\@comdblflflt`) `\@dbltoplist` にあれば上の並びで、 $p = t$, $\delta = t_1 - h$ である。なければ下の並び (`nidanfloat`, `stfloats`) で、 $p = b$, $\delta = d - r$ (r は「`\textheight - t_1`」を b で割った余り) とし、並びの先頭に `\vfil` を置く。

nidanfloat 段ごとの上のフロートは `\@cflt` と同じである。下のフロートは `\L@cflb`, `\R@cflb` を置き換え、並びを段に入る最終行の字の枠の下端にそろえ、伸びは本文とフロートの間に置く。

flushend 最終ページの段を組み上げた後、段の中のグルーを伸縮させずに組み直す（末尾に fill 次のグルー）。左右の段のつなぎ目 (`\atColsBreak`) では左段の最後の要素のベースラインから右段の最初の要素のベースラインまでを、 b の整数倍に切り上げる。フロートの箱では中身のずらし量 δ も考慮する。

evenend 最終ページの段内フロートの複製をとる前に、上のフロートを t に、下のフロートを b に組み直す。持ち越された段内フロートは `evenend` が段に入れる前に呼ぶフック `evenend/deferredfloat` で、上のフロートと同じく組み直す。

8 制限事項

- フロートの行取りは `\normalsize` の行送りが基準である。 `gyoudori` 環境と `\gyoudoriarea` も `local` をつけなければ同じである (3.1 節)。
- 箇条書きの `\topsep`, `\itemsep`, `\parsep` は行送りの整数倍ではないので、箇条書きの中や前後では行の位置がずれる。
- 縦組は考慮していない。ただし試した範囲 (pLaTeX の `tarticle` と `jlreq` の `tate`, LuaLaTeX の `ltjtarticle`) では `gyoudori` 環境、ディスプレイ数式、フロートの行取りは横組と同じように働き、前後の本文の行は行のグリッドにのる。縦組では行が右から左へ進むので、位置の `t` は右寄せ、`b` は左寄せになり、ページの先頭にきた領域は版面の右端につく。JLREQ §4.1.7 の縦組の偶数ページの例外は扱わない。縦組での 2 段組や、ほかのパッケージとの併用は試していない。
- `\flushbottom` の文書、`float` パッケージの `[H]` との組みあわせは試していない。
- `flushend` と脚注と、最終ページの下フロートまたは本文中フロートが重なると、本文が 1pt 前後ずれることがある。
- `stfloats` の `\fnbelowfloat` は pLaTeX では効かない (pLaTeX は独自の `\@makecol` を使うため)。
- `$$` で囲んだディスプレイ数式は行取りしない。
- 既知の未対応事項: `nidanfloat` は左の段からあふれた本文を右の段につなぐとき、左の段の最後の行の深さを考えずに行間を入れるので、右の段の行が「あ」の深さだけずれることがある。ページの組み方によって起きたり起きなかったりする。
- 既知の未対応事項: pLaTeX で `float`, `nidanfloat`, 脚注を一緒に使うと、段組のページの出力で `Overfull` が出ることがある。

9 参考文献

- W3C, *Requirements for Japanese Text Layout* (日本語組版処理の要件), <https://www.w3.org/TR/jlreq/>
- 阿部紀行「行取り」, <https://abenori.blogspot.com/2017/12/gyodori.html>

10 ライセンス

MIT ライセンスである。