

L^AT_EX の新クラスファイル jsarticle を拡張した 論文向けスタイルパッケージ

中鉢 欣秀¹⁾ 産技 花子²⁾

English Title

Yoshihide Chubachi¹⁾ Hanako Sangi²⁾

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis molestie ligula a diam hendrerit, id vulputate est feugiat. Suspendisse auctor neque a enim scelerisque, nec fermentum diam sodales. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Phasellus eu sodales neque. In interdum nisi at egestas aliquet. Phasellus vitae ullamcorper lectus, a commodo orci.

Keywords: AIIT, bulletin, 2013

1 はじめに

この文書では L^AT_EX の新クラスファイル jsarticle を拡張するためのスタイルファイル ajsarticle について述べる。論文のスタイルとして一般的なタイトル 1 段組, 本文 2 段組に対応している。

ajsarticle では和文に加えて英文の表題, 著者名と, 概要, キーワードを出力できるようにした。また, 明示的に「行取り」を指定できるマクロ \nlines を追加し, 二段組の左右の行が揃うことをより徹底した。

また, 本稿では日本語を主体とする文書におけるマージンの設定を geometry パッケージを利用して行う方法についても述べる。特に, 版面の高さの自動補正について解説する^{*1}。

標準的なクラスファイルに最低限の拡張を施し, より

美しく, 実用的なレイアウトを実現することを狙う。

2 コマンドの追加と再定義

2.1 英文表題

英文の表題とキーワードのために, \etitle, \eauthor, \keywords コマンドを追加した。著者名に所属を記載する場合は \thanks を使う。また, 英文の著者名で所属を参照する \thanksmark コマンドを定義した。また, 原稿の受領日を指定する \receivedon を追加した。新規に追加した \amaketitle を用いることでこれらを含むタイトルを出力できる。

2.2 行取りマクロの利用方法

行取りとは, 文字以外の図や数式を配置する際, 上下の空白を調整して行数に揃えることである。

jsarticle は基本的には自動で行取りを行なってくれるが, それでも時々ずれることがある。そこで, 明示的に行取りを指定するマクロ \nlines を作成した。

2.3 版面と余白

日本語の文書の場合, 行数と字数を設定した後, 紙面に配置をするという手順で余白 (マージン) を決めるの

Received on 2013-06-29.

¹⁾ 産業技術大学院大学/Advanced Institute of Industrial Technology

²⁾ 公立大学法人首都大学東京/Tokyo Metropolitan University

^{*1} この補正により, 脚注を用いても 行の下側が左右の段で揃う。

が望ましい．このファイルでは `geometry` パッケージの `heightrounded` オプションを用いてこの設定をしているので参考にされたい．

また、`\amaketitle` にはオプションがあり、タイトルと本文との間に \LaTeX が自動で挿入するスペースを調整できる．この値をうまく調整すれば、表題のあるページに脚注があっても、その下端を左右の段で揃えることができる．

3 制限事項

また、Abstract, Keywords:, Received on を日本語にするためのコマンドは用意していないので、スタイルファイルを直接修正して欲しい．

4 おわりに

この文書で述べた以外の図や表の挿入などは `jsarticle` の機能をそのまま利用することができる．

なお `\nlines` を実装するにあたり <http://www.dab.hi-ho.ne.jp/t-wata/tex/multicol.html> の記事を参考にした．

付録 A 脚注の確認

ここに脚注を設定する*2.

付録 B \nlines の効果

数式に対して\nlines を使用した例を以下に示す.

$$a_0 = \frac{1}{\pi} \left[\int_0^\pi a \sin kt \cdot dt + \int_\pi^{2\pi} (-a) \sin kt \cdot dt \right] = \textbf{(1)}$$

自動的に 3 行取りが行われ、数式を出力したあとでも左右の段で行が揃う. \nlines を使わないで数式を出力した例を以下に示す.

$$a_0 = \frac{1}{\pi} \left[\int_0^\pi a \sin kt \cdot dt + \int_\pi^{2\pi} (-a) \sin kt \cdot dt \right] = \textbf{(2)}$$

このとおり、左右の行がずれてしまう.

付録 C Dummey text

寿限無寿限無五劫の摺り切れ海砂利水魚の水行末雲来
末風来末. 食う寝る所に住む所藪柑子ブラコウジ. パイ
ポパイポパイポのシューリンガングーリンダイのポンポ
コピーのポンポコナーの長久命の長助.

寿限無寿限無五劫の摺り切れ海砂利水魚の水行末雲来
末風来末. 食う寝る所に住む所藪柑子ブラコウジ. パイ
ポパイポパイポのシューリンガングーリンダイのポンポ
コピーのポンポコナーの長久命の長助.

*2 当然、ここも 左右が揃う.

付録 D 版面の設計

D.1 文字サイズ

論文の原稿では、欧文の文字の大きさを 9pt とすることが多い。L^AT_EX の jsarticle クラスファイルで 9pt オプションを指定すると、欧文フォントは 9pt となるが、和文のフォントのサイズは欧文の文字に合わせるため約 8.4pt となる。

これにあわせて MS Word でも同じレイアウトを得ようとする。ところが、Word では英文と和文で異なるフォントサイズを設定することができない。加えて、フォントのサイズとして 8.4pt は指定できず、近い値として 8.5pt を用いざるを得ない。このまま L^AT_EX にあわせて行数を指定すると、和文主体の文章では少しばかり行間が詰まってしまう。

ところで、jsarticle の 12Q オプションは、計算するとフォントのサイズがほぼ 8.5pt となる。そこで、両者でフォントの大きさを揃えるならば、L^AT_EX では 12Q、Word では 8.5pt を設定するのが得策である。jsarticle の 12Q オプションはあまり広く認知されていないが、Word と共通の版面を作るにはうってつけである。

そこで、以下、文字の大きさは 12Q として、基本版面を設計していこう。

D.2 長さの単位について

L^AT_EX における $1pt = \frac{1}{72.27}inch$ である。また、 $1inch = 25.4mm$ である。また、 $1Q = 1H = 0.25mm$ である。横方向には Q を、縦方向には H を用いる。

D.3 判型・左右の余白と字詰と字間

レイアウトのベースは jsarticle に 12Q オプションを設定したものを用いている。文字のサイズは 12Q (= $3mm \approx 8.5pt$) になる。レイアウトのカスタマイズには geometory パッケージを用いた。

日本語を主体とする文書のレイアウト設計では、最初に版面（文字が印刷される領域）の大きさを求めてから、用紙に対するマージンを求めるのが一般的だ。まず、横の幅は、本文を二段組とし、1 行が 27 文字、段間が 2 文字 ($27 \times 2 + 2 = 56$ 文字) とすると、幅は $3mm \times 56$ 文字 = $168mm$ となる。

判型は一般的な A4 サイズで見開き印刷とする。段組は 2 段組で段間は 2 字とし、字間はベタ（文字と文字の

間をあけない）とする。

いま、各段の字詰を 27 字にすると、1 行あたり

$$27 \text{ 字} + 2 \text{ 字} + 27 \text{ 字} = 56 \text{ 字}$$

となる。

文字の大きさ 12Q であるから、

$$12Q \times 0.25mm \times 56 \text{ 字} = 168mm$$

が版面の横幅である。A4 の横は $210mm$ であるから、左右の余白をノド（内側） $24mm$ 、小口（外側） $18mm$ とすると、

$$168mm + 24mm + 18mm = 210mm$$

となり、ぴたりと収まる。

D.4 行間

次に、縦の長さは、行数 47 行、行間を 75%（行送り 175%）とすると、高さは $3mm \times 175\% \times 46 + 3mm = 244.5mm$ となる。行間は二分四分（文字の大きさの 75%）とする。これは jsarticle の標準の行間がほぼ二分四分であるからだが、論文で一般的な二分（50%）よりもわりとゆったりとしたものになる。ルビや圈点を入れても問題なく、タブレットなど電子媒体で読む場合でも読みやすくなるものと思われる。

行間を 75% とし文字の高さと行間とを合わせた行送り（175%）を計算すると、

$$12H \times 1.75 = 21H = 5.25mm$$

となる。

いま、行数を 47 行とすると

$$5.25mm \times 46 + 3mm = 244.5mm$$

が版面の高さとなる。A4 縦は $297mm$ であるから、

$$297mm - 244.5mm = 52.5mm$$

が余白となる。柱を入れるために天（上余白）を $30mm$ とすると、地（下余白）は $22.5mm$ となる。

実際には、jsarticle クラスで 12Q オプションを設定した場合の行送りは $16.0pt$ であり、拡大率は $\frac{1000}{1000}$ であるから、

$$16pt \times \frac{1}{72.27}inch \times 25.4mm/inch \times \frac{923}{1000} \approx 5.190mm$$

となる。これは、二分四分ほぼ近いものの、僅かな誤差がある。行数を 47 行として計算すると

$$5.190mm \times 46 + 3mm = 241.74mm$$

H で計算した場合は 244.5mm であったから、比べると 2.76mm ほど短くなる。しかし、実用上問題はない。

この版面を A4 用紙に収める場合、例えば、上の余白を多めにして天（上部）30mm，地（22.5mm），左右の余白はノドを多めに取りノド（内側）25mm，小口（外側）17mm とすればよい。

しかしながら，jsarticle の 12Q オプションを設定した場合，行送りは 175% よりもわずかに少ない。このままだと，脚注等の下側が本文よりもはみ出る格好になる。

D.5 二分あきの設定

baselinestretch に倍率を設定し，行間を二分あきにすることを検討する。

二分あきの場合，

$$12H + 12H/2 \times 0.25mm/H = 4.5mm$$

であるから，行送りを現在の $4.5mm \div 5.190mm \approx 0.867$ 倍すればよい。これを設定するには次の通りプリアンブルに記述する。

`\renewcommand{\baselinestretch}{0.8670} % 二分あきにする場合`

ただし，実際にこの設定を行ったところ，節の見出しなどで大きい文字を使うと左右の段が揃わない問題が発生する。

D.6 9pt について

jsarticle の標準である 10pt オプションでは，文字のサイズが 13Q (9.21bp) になるように調整されます。

jsarticle のオプションで 9pt の設定をすると，内部では 10pt で版組し，これを 913/1000 倍することで求める出力を得ます。

文字のサイズは 13Q に 913/1000 をかけて 11.869Q となります（これは約 8.4bp です）。

9pt オプションを設定すると行送りは 15pt になります。 $15pt \times 1/7.27 \text{ インチ} \times 25.4mm / 0.25mm = \text{約 } 21.085H$ です。これに 913/1000 をかけて 19.251H と

なりますから，行間は $19.251 - 11.869 = 7.382H$ が実際の行送りとなります。