LaTex Blog
一、图的插入
1.pdf格式的图片的插入
2.eps格式的图片的插入
2.1两个图并排
2.2三个图并排
2.3五个图并排
二、表的插入
三、数学符号
四、算法块
1.算法块的插入
五、参考文献系列
1.正文中[引用]高亮显示+超链接
2.引用文献在正文中的显示
附录：出现的问题及解决方案
1.参考文献中人名出现横线

链接：https://jiangwei99.github.io/LaTex_Blog/README.pdf

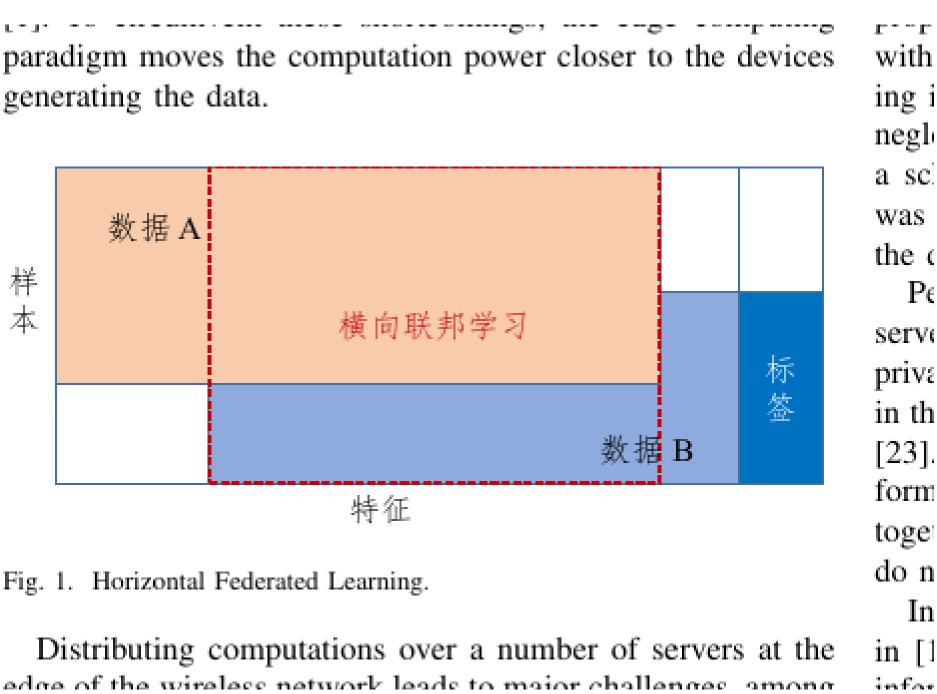
LaTex Blog

一、图的插入

1.pdf格式的图片的插入

```
\begin{figure}[htb]
    \centering
    \includegraphics[keepaspectratio, width = 1\columnwidth]{Figures/Horizontal Federated Learning.pdf}
    \caption{Horizontal Federated Learning
    \label{fig:Horizontal}}  %用于引用的标签
\end{figure}
```

效果：



[h] 表示的当前位置（here），也就是说图片排在你设置的当前位置，但是如果这一页的空间不足以放下这个图片，此时图片会转到下一页。

[t] 顶端(top)。此时系统会将图片放置在页面的顶部。

[b] 底部. (bottom) 这里是优先将图片放置在底部，也就是页面的底部。

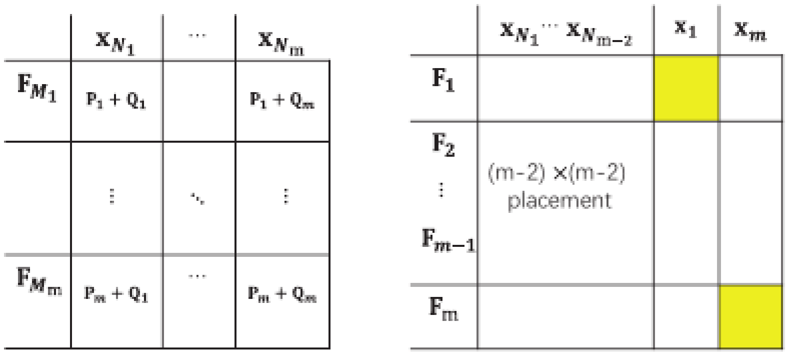
[p] 这个是将图片设置为浮动状态，也就是可以根据系统排版的，自动放置图片的位置。

2.eps格式的图片的插入

2.1两个图并排

```
\begin{figure}[!ht]
   \centerline{$\begin{array}{cc}
       \includegraphics[width=1.4in]{unsecure1.eps} &
       \hspace{-1mm}\includegraphics[width=1.6in]{unsecure2.eps}\\
       \hspace{-1mm}\mbox{\footnotesize a)    general illustration of $ \textbf{E} $. } &
       \hspace{-1mm}\mbox{\footnotesize d)    optimal placement of highlighted entries.} \\
       \end{array}$}\caption{Illustration
       of RPC.}  \label{fig.uns}
   \end{figure}
```

效果：



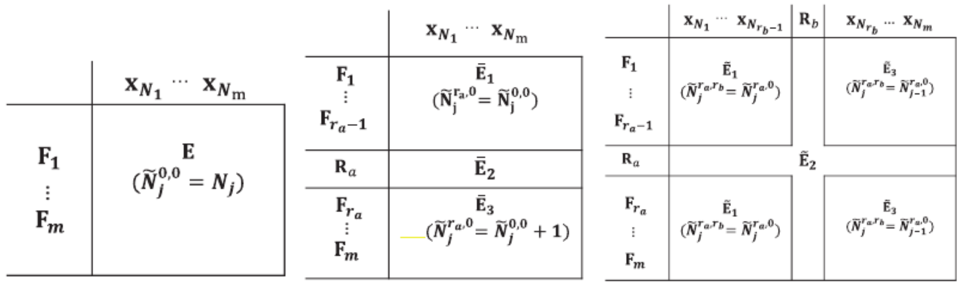
a) general illustration of **E**.　　d) optimal placement of highlighted entries.

Fig. 1. Illustration of RPC.

2.2三个图并排

```
 \begin{figure*}[!ht]
    \centerline{$\begin{array}{ccc}
        \includegraphics[width=2.0in]{secure1.eps} &
        \hspace{-1mm}\includegraphics[width=2.0in]{secure2.eps} &
        \hspace{-1mm}\includegraphics[width=2.5in]{secure3.eps}\\
        \hspace{-1mm}\mbox{\footnotesize a)    structure of
                  $ \textbf{E} $. } &
        \hspace{-1mm}\mbox{\footnotesize b)    structure of
         $ \bar{\textbf{E}} $.} &
        \hspace{-1mm}\mbox{\footnotesize c)    structure of
         $ \tilde{\textbf{E}} $.} \\
        \end{array}$}\caption{The placements of highlighted entries in SBMM.}  \label{fig.sbmm}
  \end{figure*}
```

效果：



a) structure of E.　　b) structure of $\tilde{E}$.　　c) structure of $\tilde{E}$.

Fig. 2. The placements of highlighted entries in SBMM.

2.3五个图并排

```
\begin{figure*}[!ht]
  \centerline{$\begin{array}{ccccc}
      \includegraphics[width=1.3in]{ex1.eps} &
      \hspace{-1mm}\includegraphics[width=1.3in]{ex2.eps} &
      \hspace{-1mm}\includegraphics[width=1.3in]{ex3.eps} &
      \hspace{-1mm}\includegraphics[width=1.5in]{ex4.eps} &
      \hspace{-1mm}\includegraphics[width=1.5in]{ex5.eps}\\
      \hspace{-1mm}\mbox{\footnotesize a) $ r_a = 0, r_b =0 $.} &
      \hspace{-1mm}\mbox{\footnotesize b) $ r_a = 2, r_b =0 $.}&
      \hspace{-1mm}\mbox{\footnotesize c) $ r_a = 3, r_b =0 $.}&
      \hspace{-1mm}\mbox{\footnotesize d) $ r_a = 3, r_b =5 $.} &
      \hspace{-1mm}\mbox{\footnotesize e) $ r_a = 3, r_b =4 $.} \\
      \end{array}$}\caption{Examples of the placement of highlighted entries.}  \label{fig.ex}
  \end{figure*}
```

效果：

Fig. 2. The placements of highlighted entries in SBMM.

	x_2	x_3	x_1	x_4
F_1	0	2	6	7
F_2	1	3	7	8
F_3	2	4	8	9
F_4	3	5	9	10

a) $r_a = 0, r_b = 0$.

	x_2	x_3	x_1	x_4
F_1	0	3	8	9
R_A	1	4	9	10
F_2	2	5	10	11
F_3	3	6	11	12
F_4	4	7	12	13

b) $r_a = 2, r_b = 0$.

	x_2	x_3	x_1	x_4
F_1	0	2	7	8
F_2	1	3	8	9
R_A	2	4	9	10
F_3	3	5	10	11
F_4	4	6	11	12

c) $r_a = 3, r_b = 0$.

	x_2	x_3	x_1	x_4	R_B
F_1	0	2	7	8	13
F_2	1	3	8	9	14
R_A	2	4	9	10	15
F_3	3	5	10	11	16
F_4	4	6	11	12	17

d) $r_a = 3, r_b = 5$.

	x_2	x_3	x_1	R_B	x_4
F_1	0	2	7	8	9
F_2	1	3	8	9	10
R_A	2	4	9	10	11
F_3	3	5	10	11	12
F_4	4	6	11	12	13

e) $r_a = 3, r_b = 4$.

Fig. 3. Examples of the placement of highlighted entries.

二、表的插入

三、数学符号

ℓ `\[\ell \]`

$\mathcal L$ `\mathcal L`

$\mathcal N$ `$\mathcal N$`

四、算法块

算法块需要调用包

```
\usepackage[ruled,linesnumbered]{algorithm2e}
```

1.算法块的插入

```
\begin{algorithm}\label{alg1}
    \LinesNumbered
    \begin{small}
        \KwIn{$ N $, $ r_a $, $ r_b $, $ M $, $ P $}
        \KwOut{$ Q $, $ Q_b $}
        $ Q_1 = 0 $;\\
        \For{$ j \ge 2 $; $j < r_b $; $j=j+1$}{
            \If {$ m+1 - \tilde{N}_{j-1}^{r_a,r_b}  \le \tilde{N}_{j}^{r_a,r_b}-1 $}{
                $ Q_j = Q_{j-1} + \tilde{N}_{j-1}^{r_a,r_b} $;\\
            }
            \Else
            { $ Q_j = Q_{j-1} +m - \tilde{N}_{j}^{r_a,r_b}+2 $;\\}
        }
        $ Q_{b} = Q_{r_b - 1} + \tilde{N}_{r_b-1}^{r_a,r_b} $;\\
        $ Q_{r_b} = Q_{b} +m - \tilde{N}_{r_b+1}^{r_a,r_b}+2 $;\\
        \For {$ j \ge r_b + 1 $; $ j \le m $; $ j=j+1 $ }{
            \If {$ m+1 - N_{j}^{r_a,r_b}  \le N_{j+1}^{r_a,r_b}-1 $}{
                $ Q_j = Q_{j-1} + N_{j}^{r_a,r_b} $;\\
            }
            \Else
            { $Q_j = Q_{j-1} +m - N_{j+1}^{r_a,r_b}+2 $;\\}
        }

        \Return  $ Q $, $ Q_b $ ;
        \caption{The Optimal Values of $ Q $ with a Given Placement of Highlighted Entries.} \label{optimal}
    \end{small}
\end{algorithm}
```

效果：

Algorithm 1: The Optimal Values of Q with a Given Placement of Highlighted Entries.

Input: N, r_a, r_b, M, P
Output: Q, Q_b
$Q_1 = 0$;
for $j \geq 2$; $j < r_b$; $j = j + 1$ **do**
  if $m + 1 - \tilde{N}_{j-1}^{r_a,r_b} \leq \tilde{N}_{j}^{r_a,r_b} - 1$ **then**
    $Q_j = Q_{j-1} + \tilde{N}_{j-1}^{r_a,r_b}$;
  end
  else
    $Q_j = Q_{j-1} + m - \tilde{N}_{j}^{r_a,r_b} + 2$;
  end
end
$Q_b = Q_{r_b-1} + \tilde{N}_{r_b-1}^{r_a,r_b}$;
$Q_{r_b} = Q_b + m - \tilde{N}_{r_b+1}^{r_a,r_b} + 2$;
for $j \geq r_b + 1$; $j \leq m$; $j = j + 1$ **do**
  if $m + 1 - N_{j}^{r_a,r_b} \leq N_{j+1}^{r_a,r_b} - 1$ **then**
    $Q_j = Q_{j-1} + N_{j}^{r_a,r_b}$;
  end
  else
    $Q_j = Q_{j-1} + m - N_{j+1}^{r_a,r_b} + 2$;
  end
end
return Q, Q_b ;

tomer has its preferences, which are encoded by an attribute vector x. Based on a customer's preferences, the application

comp latenc scenar the la [16], This decod of a l rely c privac Furthe the va linear increa *No* upper vector vector finite We u $\{1, 2,$ than c than notati $3 \mapsto$

五、参考文献系列

1.正文中[引用]高亮显示+超链接

```
1  \usepackage[backref]{hyperref}   %用于引用突出，会报错，但能编译
```

效果如图

B. Coded Computing

There are two main challenges in distributed computing: How to reduce communication overhead and ensure safety. In order to calculate the final result, a large number of intermediate results need to be exchanged between computing nodes through the network. This not only increases communication overhead, but also limits the performance of distributed computing applications. For example, for the Hadoop cluster on Facebook, the average time spent in the data transfer phase accounts for 33% of the entire job execution time [4]. When running the Terasort application on a heterogeneous Amazon EC2 cluster, data shuffling will take 65% of the time. Approximately 70% of the total execution time is spent on the Self-Join application [5]. In the actual training of convolutional neural networks, communication restrictions are even more embarrassing. As we all know, millions of model parameters need to be updated in ResNet-50 [6] and AlexNet [7].

[16] Q. Yu, M. A. Maddah-Ali, and A. S. Avestimehr, "Straggler mitigation in distributed matrix multiplication: Fundamental limits and optimal coding," *IEEE Transactions on Information Theory*, vol. 66, no. 3, pp. 1920–1933, 2020. II-A
[17] L. Chen, H. Wang, Z. Charles, and D. Papailiopoulos, "DRACO: Byzantine-resilient distributed training via redundant gradients," in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. PMLR, 10–15 Jul 2018, pp. 903–912. [Online]. Available: https://proceedings.mlr.press/v80/chen18l.html II-A, III
[18] W.-T. Chang and R. Tandon, "On the capacity of secure distributed matrix multiplication," in *2018 IEEE Global Communications Conference (GLOBECOM)*, 2018, pp. 1–6. II-B, II-B, II-B
[19] R. Bitar, P. Parag, and S. El Rouayheb, "Minimizing latency for secure distributed computing," in *2017 IEEE International Symposium on Information Theory (ISIT)*, 2017, pp. 2900–2904. II-B, II-B
[20] R. Bitar, P. Parag, and S. El Rouayheb, "Minimizing latency for secure coded computing using secret sharing via staircase codes," *IEEE Transactions on Communications*, vol. 68, no. 8, pp. 4609–4619, 2020. II-B

2.引用文献在正文中的显示

参考文献引用出现这种情况：

thereby reducing co
[9], [15] mainly
which has limitatio

导入包：

```
\usepackage[numbers,sort&compress]{natbib} %用于解决[9],[15]，把变成[9,15]
```

效果如下：

thereby reducing c
[9, 15] mainly
which has limitati

附录：出现的问题及解决方案

1.参考文献中人名出现横线

trix multiplication," in *2018 IEEE Global Communications Conference (GLOBECOM)*, 2018, pp. 1–6.
[19] R. Bitar, P. Parag, and S. El Rouayheb, "Minimizing latency for secure distributed computing," in *2017 IEEE International Symposium on Information Theory (ISIT)*, 2017, pp. 2900–2904.
[20] ——, "Minimizing latency for secure coded computing using secret sharing via staircase codes," *IEEE Transactions on Communications*, vol. 68, no. 8, pp. 4609–4619, 2020.

原因：这是因为相邻两篇文献的作者相同，在IEEE模板下会出现横线。

找到TexLive安装目录下的**"IEEEtran.bst"**文件

```
路径：D:\Program Files\TexLive\2021\texmf-dist\bibtex\bst\IEEEtran.bst
```

找到如下代码(大约在128行)，将代码中的#1**变为**#0

```
FUNCTION {default.is.dash.repeated.names} { #1 }
```

重新编译，解决问题

(*GLOBECOM*), 2018, pp. 1–6.

[19] R. Bitar, P. Parag, and S. El Rouayheb, “Minimizing latency for secure distributed computing,” in *2017 IEEE International Symposium on Information Theory (ISIT)*, 2017, pp. 2900–2904.

[20] R. Bitar, P. Parag, and S. El Rouayheb, “Minimizing latency for secure coded computing using secret sharing via staircase codes,” *IEEE Transactions on Communications*, vol. 68, no. 8, pp. 4609–4619, 2020.

[21] J. Kakar, S. Ebadifar, and A. Sezgin, “Rate-efficiency and straggler-