声波成像—叠加成像（CMP Stacking）

问题描述

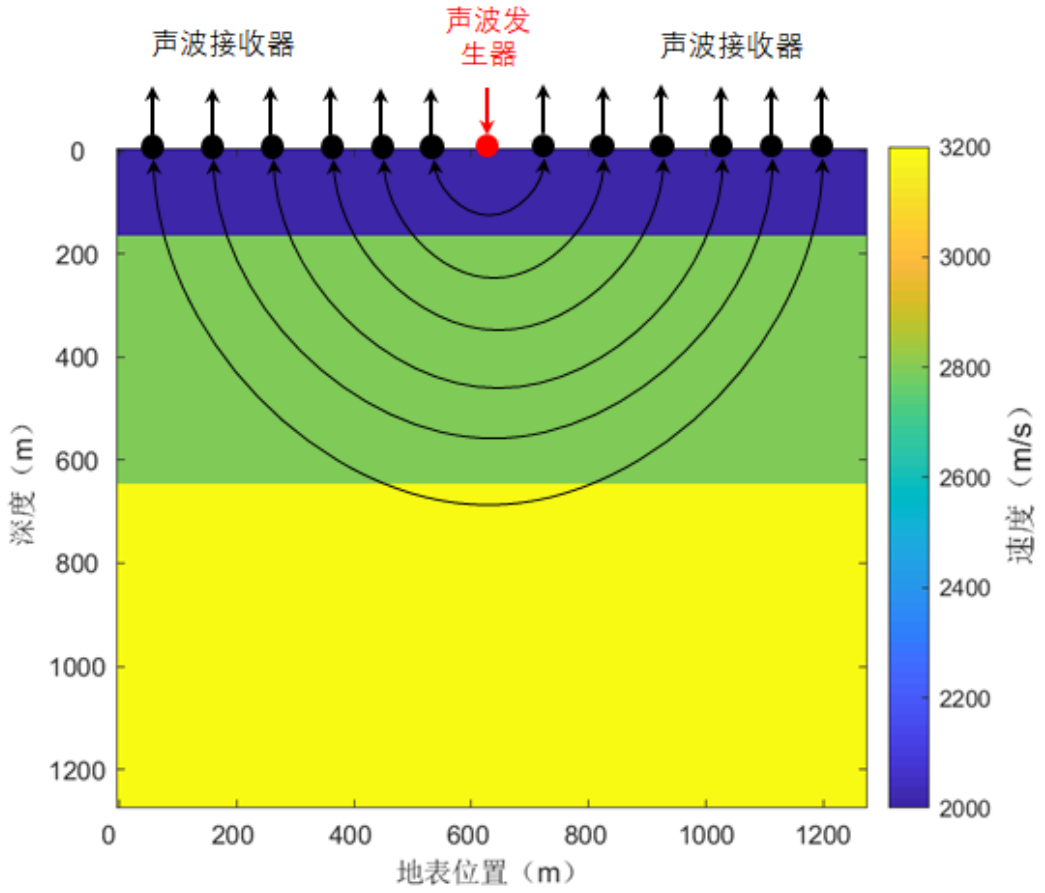
- Wave Propagation部分对声波传播过程进行建模，得到声波信号在地下传播过程中形成的波场：$\psi(x,z,t)$
- 并得到波动方程：

$$\nabla^2\psi(x,z,t)=\frac{1}{v(x,z)^2}\frac{\partial^2\psi(x,z,t)}{\partial t^2}$$

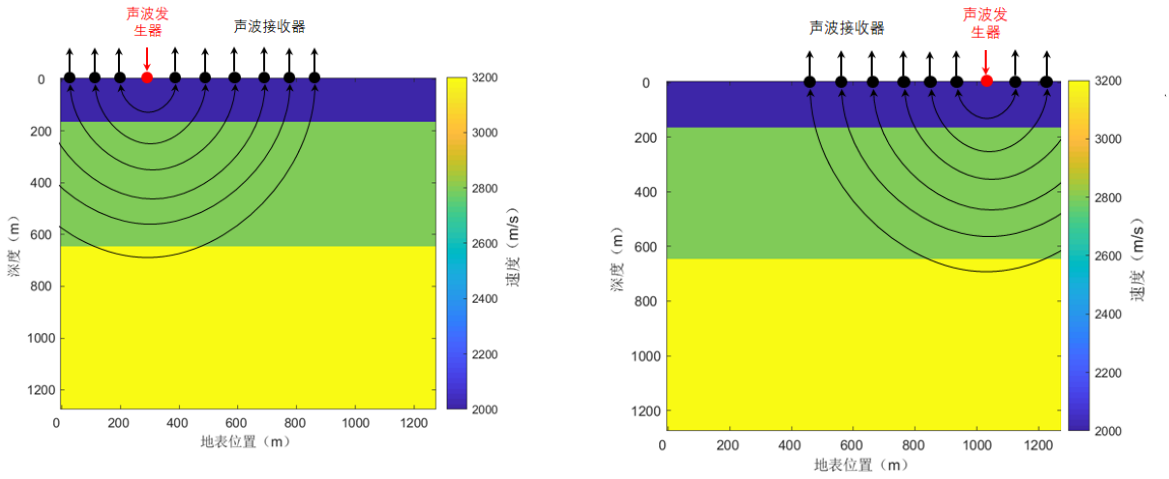
    - 其中，$v(x,z)$表示不同位置的速度，包含地下地质结构信息
- 根据$v(x,z)$生成得到波场$\psi(x,z,t)$的过程称为正演过程（Forward Modelling）
- 反之，根据观测得到的波场$\psi(x,z,t)$反推地下模型$v(x,z)$的过程称为反演过程（Inverse Problem），声波成像的主要目的是实现从波场数据到速度模型的反演过程
- 叠加成像是声波成像的重要步骤之一，通过叠加成像的方法，可以将Wave Propagation得到的波场数据整理成与地下地质结构对应的二维数据，进而在此二维数据基础上得到地下速度模型

数据准备

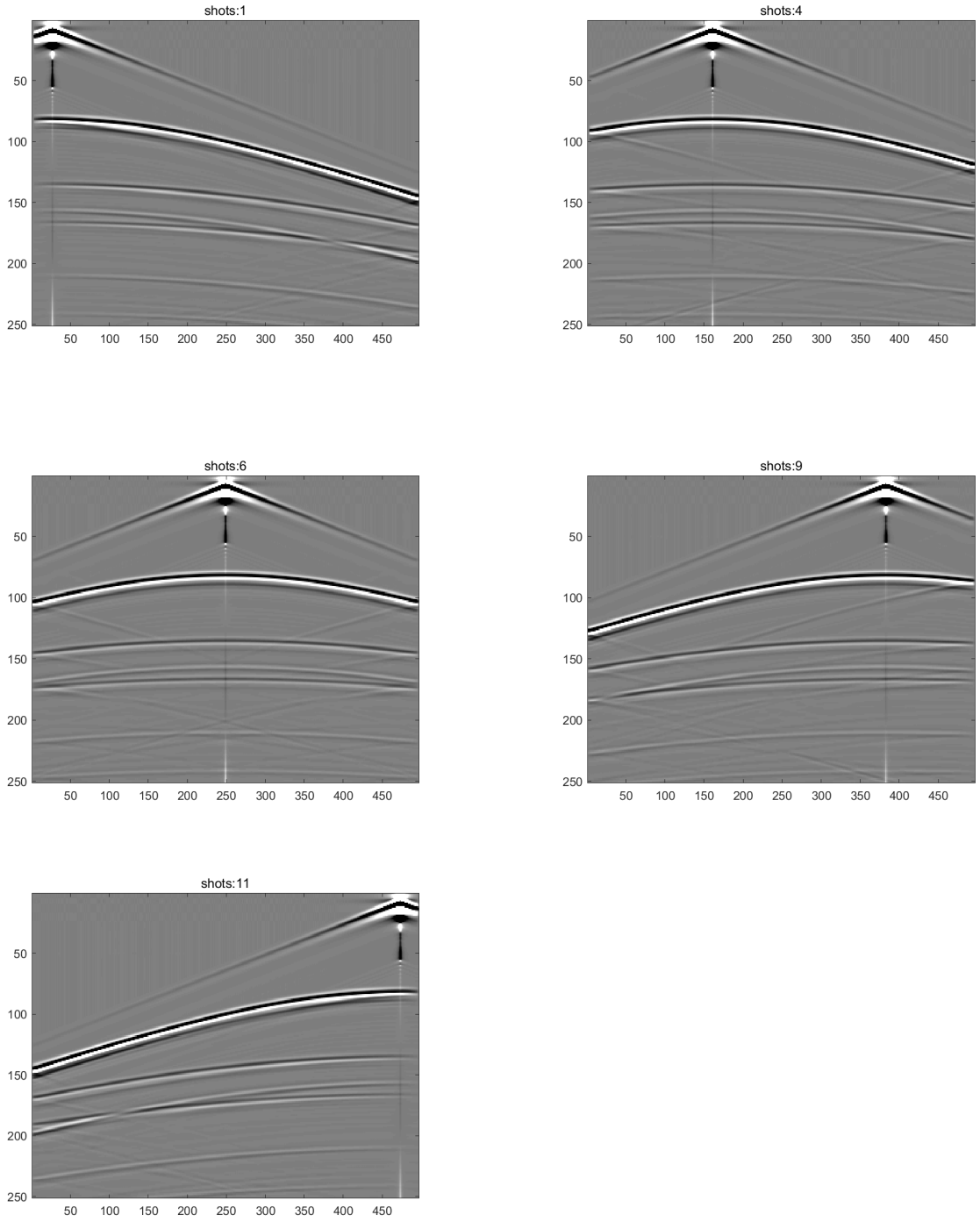
- 如Wave Propagation部分所述，可以通过一个发生器激发声波，多个接收器接收声波的方式得到对地下波场的观测：



- 根据声波在传播过程中能量逐渐衰减的原理可知，离声波发生器越近的位置，声波能量越强，成像效果越好，反之，越远的位置，声波能量越弱，成像效果越差
- 为提升不同位置的声波成像效果，可以将声波发生器放置在不同位置，通过多次激发声波，多次接收声波的方式，得到对地下波场的多次观测，再叠加多次观测的结果，得到最终的成像结果：



- 将声波发生器均匀放在11个位置，接收器接收得到第1、4、6、9、11个声波发生位置的波场数据如下：



- 其中，第6个位置成像过程的声波传播过程、接收器信号接收过程如下所示：
- 基于上述11次激发的数据（大小为$11 \times T \times X$），T表示时间方向的接收点数，X表示声波接收器数量，进行叠加成像

CMP Stacking方法原理

- 基本目标是对上述得到的三维数据进行整理、叠加，得到二维成像结果
- 11次声波发生、接收点的位置示意图如下所示：

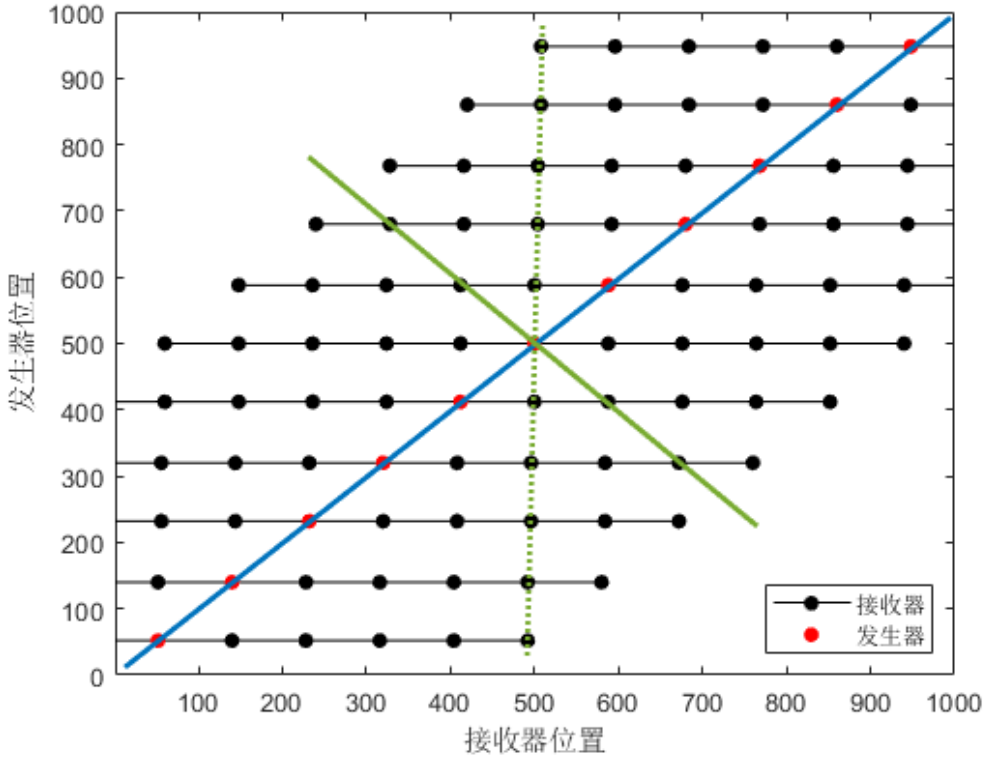
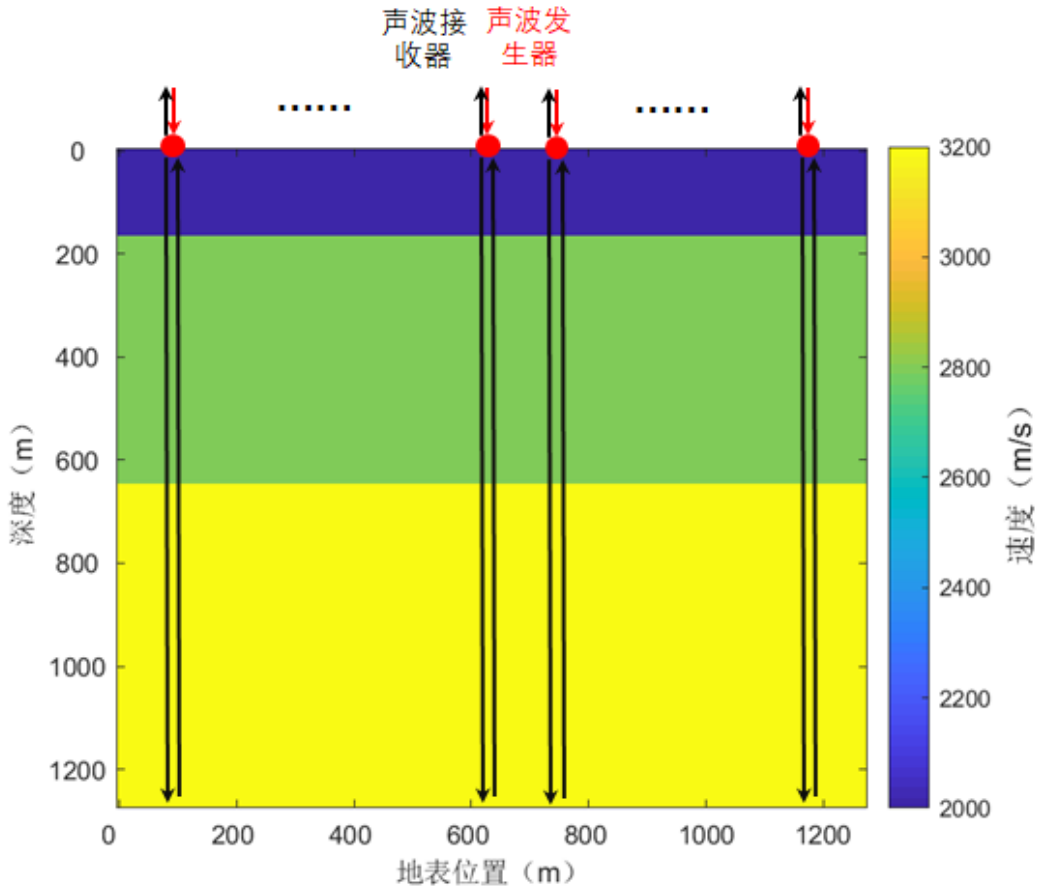


图1 声波发生器、接收器位置示意图

- 每一行分别表示一次发生接收，每一行的红色点表示声波发生器位置，黑色点表示声波接收器位置

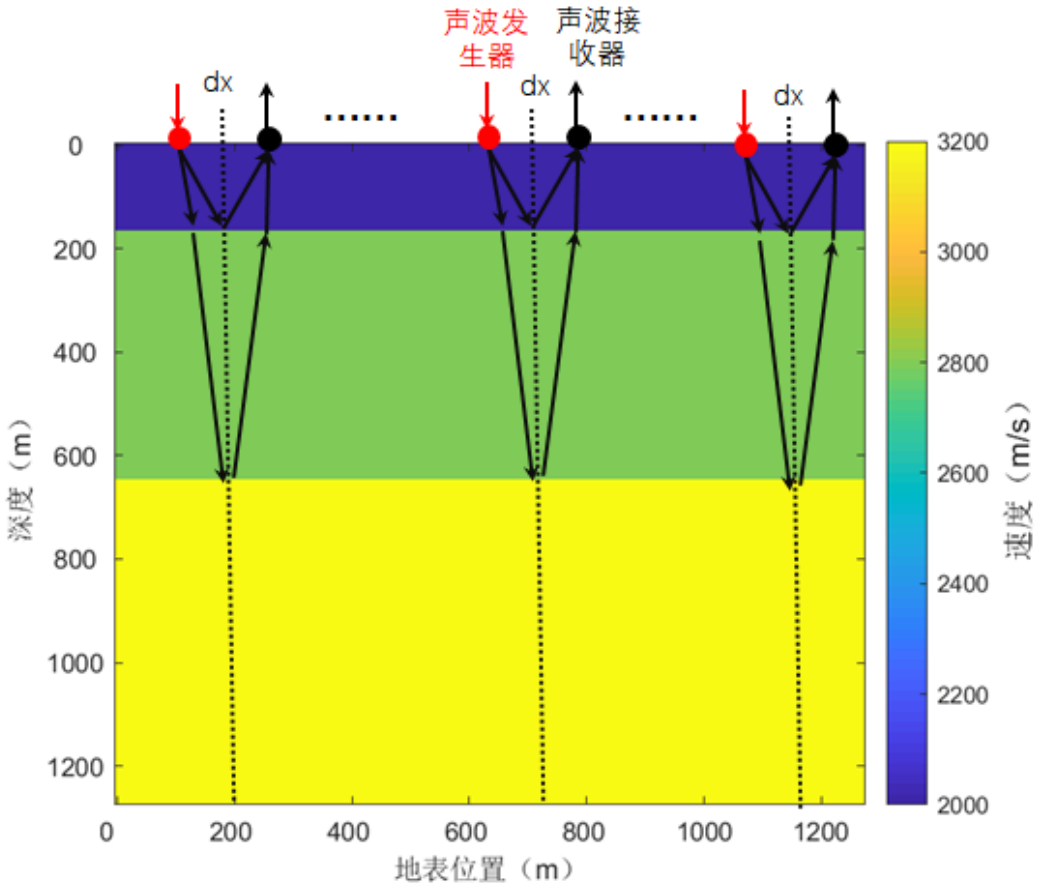
共偏移距数据（Common Offset）

- 若提取所有红色点位置的声波接收结果，表示在11个位置激发声波，并在该11个位置接收声波，如下所示：



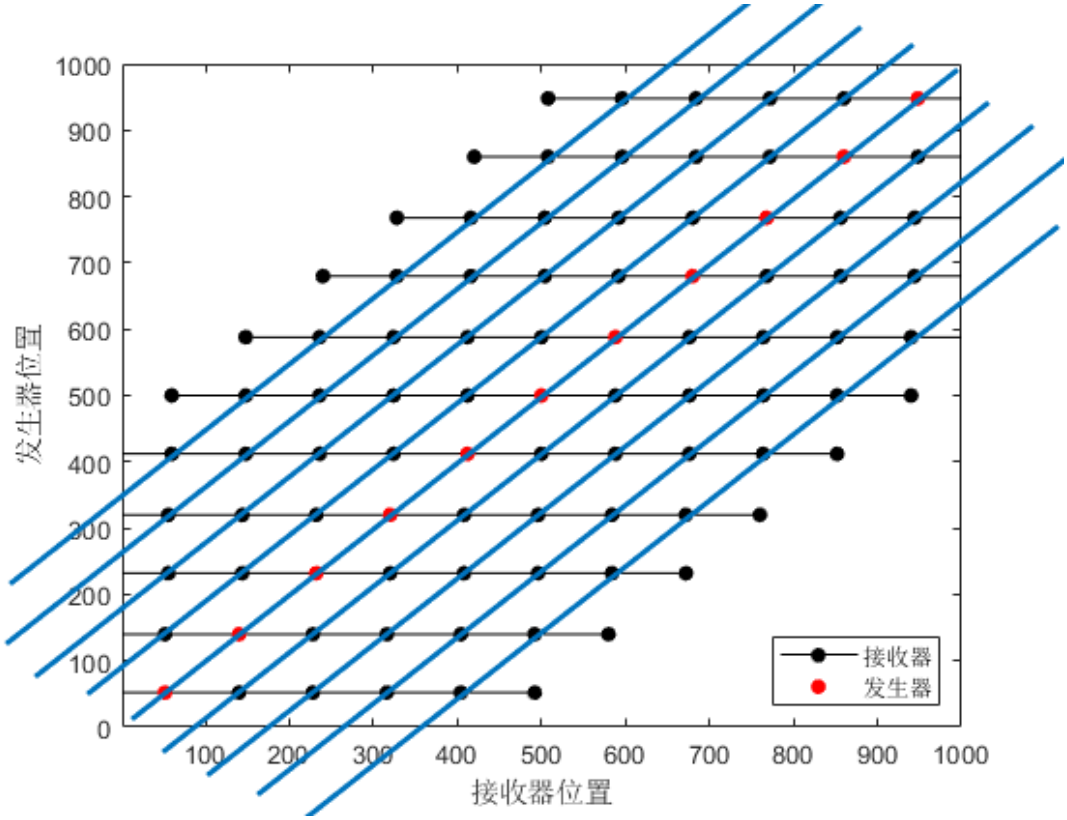
  - 可以得到一个$T \times 11$大小的声波接收数据，相当于对11个列的位置进行采样，得到横向分辨率为11的地下成像结果

- 上述红色点位置称为偏移距为0的数据（偏移距：接收器和发生器的距离）
- 类似的，可以从图1中提取偏移距相同且不为零的数据，比如：



    - 相当于对虚线所在列进行成像，由此又得到一个横向分辨率为11的成像结果

- 每11个共偏移距数据，构成一个横向分辨率为11的成像结果，那么，把所有这样的数据结合起来，就可以得到横向分辨率较高的成像结果
- 结合图1，一组共偏移距数据即对应与图中蓝色实线平行的接收器组，可以从图1中提取多组共偏移距数据：



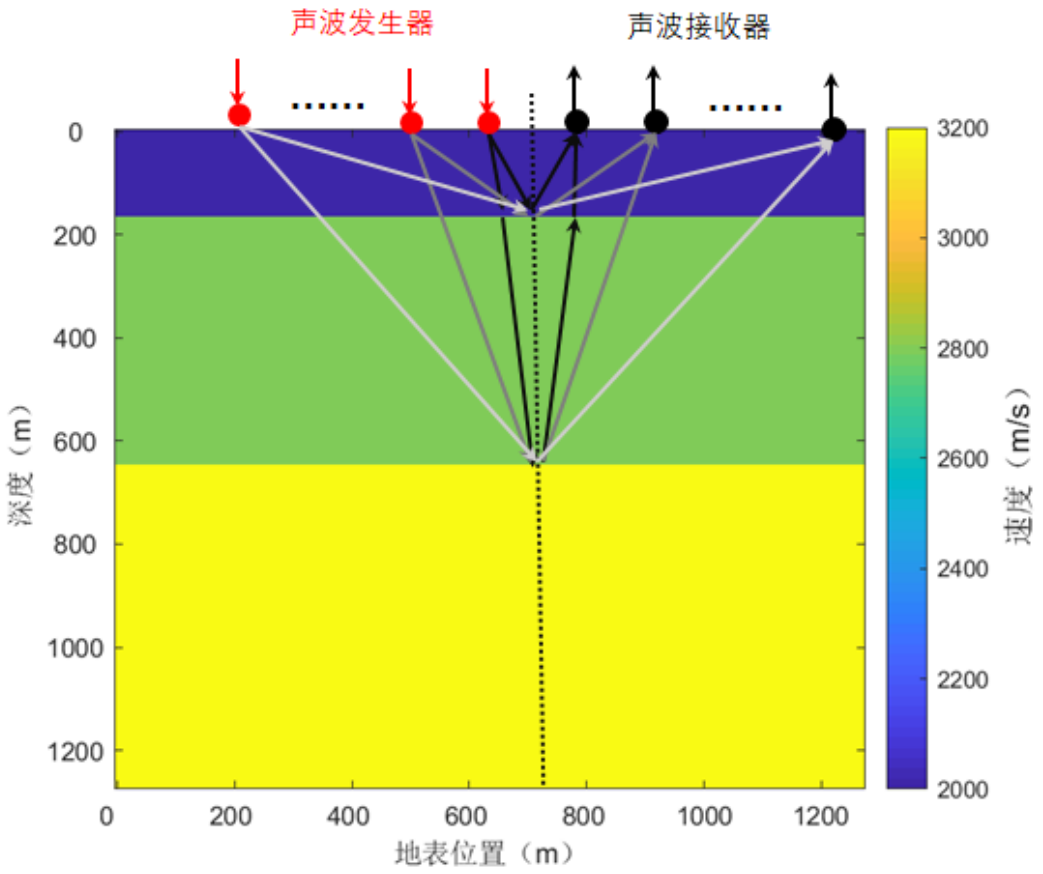
- 易得根据接收器、发生器坐标位置计算偏移距的计算公式如下：

$$xoff = \mathrm{abs}(xrec - xshot)$$

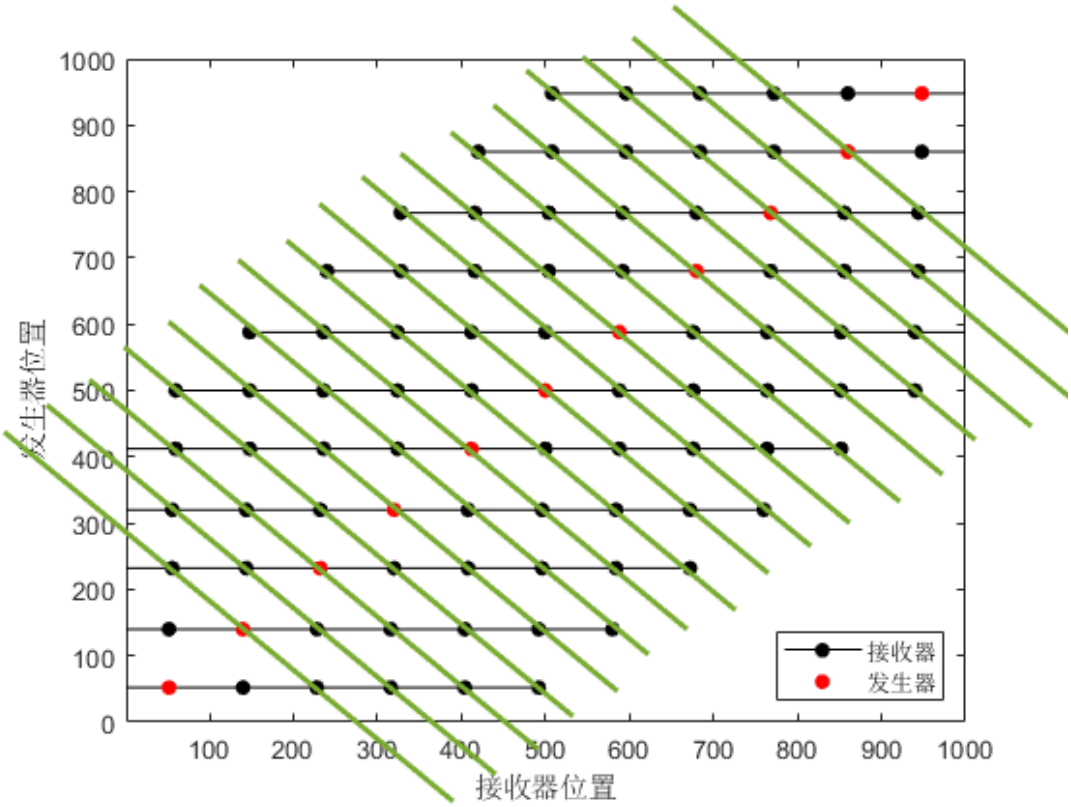
其中，$xoff$表示中心点坐标，$xrec$表示接收器位置，$xshot$表示发生器位置，$\mathrm{abs}(\cdot)$表示绝对值

共中心点数据（Common Midpoint， CMP）

- 共中心点数据表示对同一列进行成像：



- 可以简单理解为，上图中的声波接收器都是对同一列（图中虚线）所示位置进行成像的结果，即表示的是同样的信息，只不过离中心点近的接收器接收的信号比较早，离中心点远的接收器接收的信号比较晚，对接收延迟进行矫正之后，可以对以上声波接收器接收到的信号进行叠加（Stacking），增强有效信号，减弱噪声，得到对上图虚线列所示位置的更好的成像结果
- 结合图1，一组共中心点数据即对应与图中绿色实线平行的接收器组，可以从图1中提取多组共中心点数据：



- 易得，根据接收器位置、发生器坐标位置计算中心点坐标的计算公式为：

$$xcmp = \frac{xrec + xshot}{2}$$

其中，$xcmp$表示中心点坐标，$xrec$表示接收器位置，$xshot$表示发生器位置

叠加过程

- CMP叠加过程本质上即对数据沿绿线方向投影到蓝线上，对于投影到同一点的数据进行延时矫正后叠加，得到该位置的成像结果，组合起来，得到最终的二维成像结果

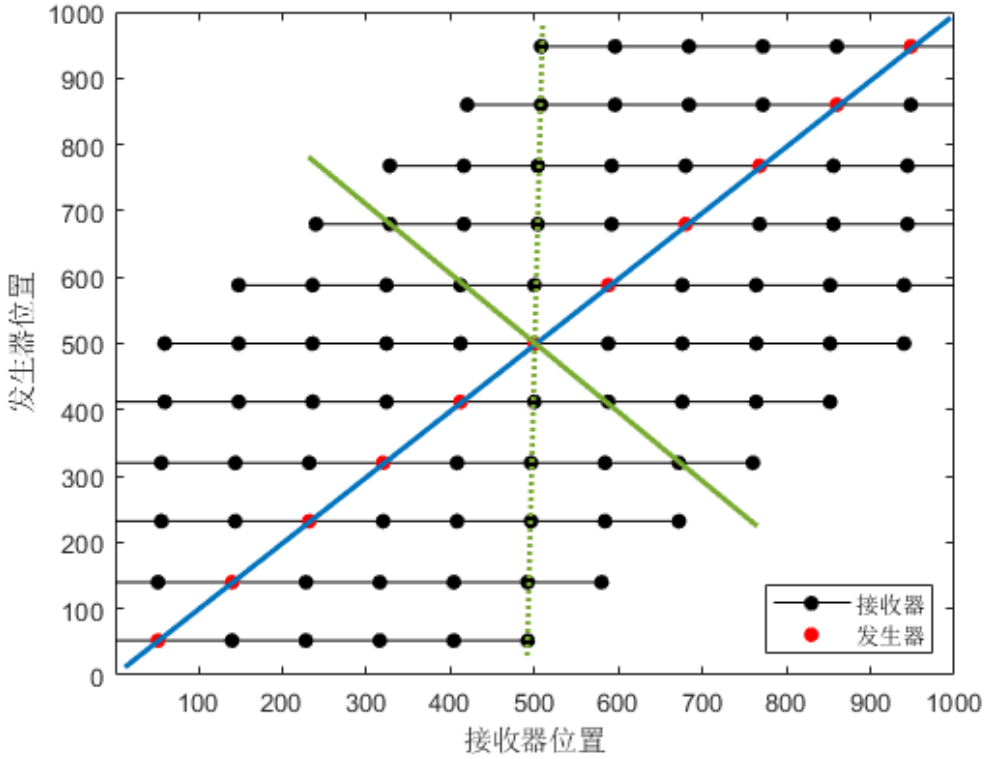


图1 声波发生器、接收器位置示意图

Matlab实现

- 代码：

https://github.com/kiyoxi2020/sonic-imaging

- 放置11个发生器，模拟声波传播、接收过程：

```
vel = vmodel;
nshots=11;
fdom=30;
[w,tw]=wavemin(dt,fdom,.2);%source waveform
[shots,t,xshots,xrecs,shotnames]=afd_shootline(dx,vel,dt,dtstep,w,tw,tmax,nshots);
save(['vmodel_shots',num2str(nshots),'.mat'], 'shots', 't', 'xshots', 'xrecs', 'shotnames');
```

- 滤除不需要的声波信号（比如未经过反射，直接沿地表传播的波）：

```
%% 预处理：滤除不需要的声波信号
load(['vmodel_shots',num2str(nshots),'.mat'])
xoffref=500;              % 设置滤波参考偏移距
```

```
xmute=[0 xoffref];       % 滤波偏移距范围
tmute0=.22;              % 零偏移距位置的滤波时间点
tmute1=.4;               % 参考偏移距位置的滤波时间点
tmute=[tmute0 tmute1];   % 滤波偏移距范围对应的时间范围
shotsg=preprocess_seis(shots,t,xrecs,xshots,xmute,tmute)
```

- 对每个发生-接收数据进行投影、延时矫正和叠加：

```
for k=1:nshots
    % 将发生、接收点坐标投影到共中心点、共偏移距坐标，并进行延时矫正
    [shot_nmo,~,~,~]=compute_nmor_cmp(shots{k},t,xrec{k},xshots(k),...
        velrms,xv,cmp(1),cmp(2),cmp(3));
    if(k==1)
        % 初始化成像数据
        stack=zeros(size(shot_nmo));
        foldstack=stack;
    end
    % 对共中心点数据进行叠加
    stack=stack+shot_nmo;
    %记录每个点的叠加次数
    shot_fold=ones(size(shot_nmo));
    ind=shot_nmo==0;
    shot_fold(ind)=0;
    foldstack=foldstack+shot_fold;
end
% 根据叠加次数进行归一化
ind=find(foldstack~=0);
stack(ind)=stack(ind)./foldstack(ind);
```

- 坐标投影过程计算如下：

```
% 计算中心点坐标
xcmpraw=(x+xshot)/2;
% 计算偏移距
xoff=abs(x-xshot);
```

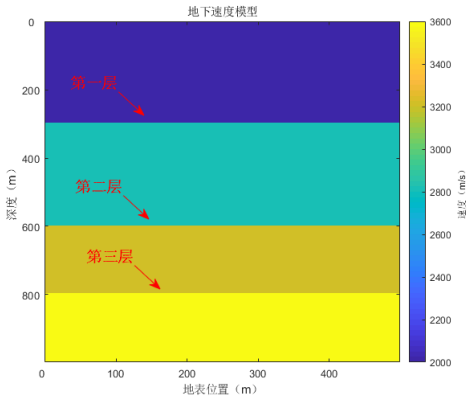
- 获取初始速度，并根据初始速度进行延时矫正：

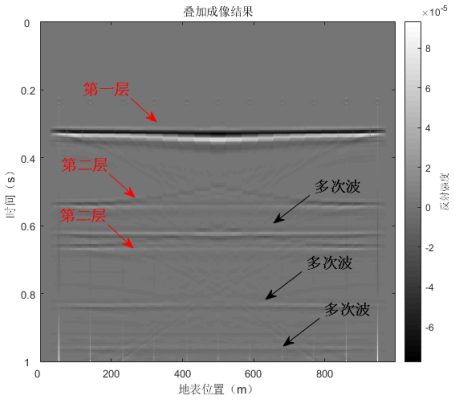
```
% 插值计算该列对应的初始速度
vrms=(xcmpraw(k)-xv(ind(1)+1))*vrmsmod(:,ind(1))/(xv(ind(1))-xv(ind(1)+1))...
    +(xcmpraw(k)-xv(ind(1)))*vrmsmod(:,ind(1)+1)/(xv(ind(1)+1)-xv(ind(1)));

% 根据偏移距，计算延时
tx=sqrt( (xoff(k)./vrms).^2 + t.^2 );

% 应用sinc函数插值，进行延时矫正
ind=between(t(1),t(end),tx,2);
sout(ind)=sinci(shot(:,k),t,tx(ind))';
```

- 得到成像结果：


地下速度模型
第一层
第二层
第三层
深度（m）
地表位置（m）
速度（m/s）


叠加成像结果
第一层
第二层
第二层
多次波
多次波
多次波
时间（s）
地表位置（m）