

```

133
134
135
136
137
138
139
140
141
142
143
144
145
146
147

```

`self.Interpolayer_uu = nn.Linear(self.mp-2,1,bias=False)
self.NodalValues = nn.Parameter(data=torch.ones(1))
self.NodalValues_uu = nn.Parameter(data=torch.ones(1))
self.NodalValues_uu = nn.Parameter(data=torch.ones(1))
self.Interpolayer_uu.weight.data = self.Nodal
self.Interpolayer.weight.requires_grad =
self.Functions_dd = nn.ModuleList([
self.Interpolayer_dd = nn.Linear(2,1)
self.Interpolayer_dd.weight.requires_grad =
self.SumLayer = nn.Linear(2,1,bias=False)
self.SumLayer.weight.data.fill_([1,1])
self.SumLayer.weight(0,1) = 1
self.SumLayer.weight.requires_grad = True
def forward(self,x):
 intermediate_uu = self.Interpolayer_uu(x)
 intermediate_uu = self.NodalValues_uu * intermediate_uu
 out_uu = self.SumLayer(intermediate_uu)
 out_uu = self.Interpolayer_dd(out_uu)
 u = out_uu`



IP PARIS