

Passo a Passo — Executar MPI (mpi4py) em Múltiplas Máquinas (Windows, macOS e Linux)

1) Visão Geral

Este guia mostra, passo a passo, como executar programas Python com mpi4py distribuídos em várias máquinas da sua rede local.

Cobriremos Linux/macOS (Open MPI/MPICH) e Windows (MS-MPI), incluindo pré-requisitos, configuração de hosts, execução e diagnóstico.

2) Pré-requisitos (todas as máquinas)

- Uma implementação de MPI compatível: Open MPI ou MPICH (Linux/macOS) | MS-MPI (Windows).
- Python 3 + mpi4py instalados no mesmo caminho/ambiente (recomenda-se venv/Conda).
- Acesso de rede entre os nós (SSH no Linux/macOS; SMPD e credenciais no Windows).
- Código acessível em todas as máquinas (pasta compartilhada OU cópia idêntica no mesmo caminho).

3) Teste mínimo (hello_mpi.py)

Crie um arquivo hello_mpi.py com o conteúdo abaixo (idêntico em todas as máquinas ou em uma pasta compartilhada).

Código:

```
from mpi4py import MPI
import socket
comm = MPI.COMM_WORLD
print(f"Hello from rank {comm.rank}/{comm.size} on {socket.gethostname()}")
```

4) Linux/macOS — Open MPI (ou MPICH)

4.1 Instale o MPI e o mpi4py em todas as máquinas:

```
sudo apt-get install openmpi-bin libopenmpi-dev # (Debian/Ubuntu)
```

ou baixe/instale MPICH conforme sua distro

pip install mpi4py

4.2 Configure SSH sem senha (no nó lançador):

ssh-keygen -t ed25519 -N "" -f ~/.ssh/id_ed25519

ssh-copy-id usuario@node1

ssh-copy-id usuario@node2

ssh usuario@node1 hostname # teste

4.3 Crie um hostfile (hosts.txt) no nó lançador:

node1 slots=8

node2 slots=8

(Ajuste 'slots' ao número de núcleos/recursos de cada nó.)

4.4 Execute via Open MPI (a partir do diretório do projeto):

```
mpirun -np 8 --hostfile hosts.txt \  
-x PATH -x LD_LIBRARY_PATH -x PYTHONPATH \  
python hello_mpi.py
```

Dicas (Open MPI):

- Se houver múltiplas interfaces de rede:

mpirun --mca btl_tcp_if_include eth0 ...

- Forçar TCP puro (evita RDMA/Infiniband):

mpirun --mca pml ob1 --mca btl tcp,self ...

- Executar em diretório remoto específico:

mpirun --wdir /caminho/no/no ...

4.5 MPICH (varia por empacotamento, exemplo geral):

```
mpirun -f hosts.txt -n 8 python hello_mpi.py
```

5) Windows — MS-MPI

5.1 Instale MS-MPI (runtime + SDK) em todas as máquinas; instale Python + mpi4py:

```
pip install mpi4py
```

5.2 Inicie o serviço SMPD (como Administrador, em cada máquina):

```
smpd -install
```

```
smpd -start
```

Libere o smpd.exe no Firewall.

5.3 Registre credenciais (no nó lançador):

```
mpiexec -register
```

(ou use -account e -password na linha de comando)

5.4 Especifique hosts e execute:

Lista inline:

```
mpiexec -n 8 -hosts 2 HOSTA HOSTB python hello_mpi.py
```

Machinefile (hosts.txt):

```
HOSTA
```

```
HOSTB
```

```
mpiexec -n 8 -machinefile hosts.txt python hello_mpi.py
```

5.5 Diretório compartilhado (UNC):

```
mpiexec -wdir \\SERVIDOR\projeto -n 8 -hosts 2 HOSTA HOSTB python hello_mpi.py
```

6) Sincronização e ambiente

6.1 Sincronizar início (comparação justa de tempos):

```
from mpi4py import MPI

comm = MPI.COMM_WORLD

comm.Barrier()

t0 = MPI.Wtime()
```

6.2 Exportar variáveis de ambiente (Open MPI):

```
mpirun -np 8 --hostfile hosts.txt -x PATH -x LD_LIBRARY_PATH -x PYTHONPATH ...
```

6.3 Ambientes virtuais:

- Use o MESMO caminho de venv/Conda em todos os nós, ou exporte PATHs adequados.
- Alternativa: um wrapper remoto que ativa a venv e chama seu script.

7) Organização do código entre nós

Opção A — Pasta compartilhada (NFS/Samba/SMB/UNC):

- Monte/acesse a mesma pasta em todos os nós e execute a partir dela.

Opção B — Cópia idêntica:

- Garanta o mesmo caminho absoluto (ex.: /opt/projeto ou C:\projetos\app) e atualize tudo em conjunto.

8) Exemplos de execução reais

Open MPI (Linux/macOS):

```
mpirun -np 16 --hostfile hosts.txt -x PATH -x LD_LIBRARY_PATH \

python <SOURCE>/mpi_trapezio_coletiva.py
```

MPICH (Linux/macOS):

```
mpirun -f hosts.txt -n 16 python <SOURCE>/mpi_isend_crianças.py
```

MS-MPI (Windows):

```
mpiexec -n 16 -machinefile hosts.txt python <SOURCE>\mpi_funcoes_crianças_gather.py
```

MS-MPI com UNC:

```
mpiexec -wdir \\SERVIDOR\projeto -n 16 -hosts 2 WIN1 WIN2 \  
python <SOURCE>\mpi_bsend_crianças.py
```

9) Diagnóstico rápido (problemas comuns)

- Não conecta: verifique SSH (Linux/macOS) ou SMPD/credenciais (Windows).
- Biblioteca não encontrada: ajuste PATH/LD_LIBRARY_PATH; exporte com -x (Open MPI).
- Travou/timeout: libere Firewall; fixe interface com --mca btl_tcp_if_include; teste ping.
- Versões diferentes de MPI/mpi4py: padronize.
- Arquivo não encontrado: confirme --wdir e compartilhamento.
- Permissões: confira credenciais/ACLs no compartilhamento.

10) Checklist final

- ☐ Mesmas versões de MPI e mpi4py em todos os nós
- ☐ Python/venv/Conda no mesmo caminho (ou PATH exportado)
- ☐ Código acessível (pasta compartilhada ou cópia idêntica)
- ☐ SSH sem senha (Linux/macOS) OU SMPD ativo e credenciais registradas (Windows)
- ☐ hostfile/machinefile preparado e testado
- ☐ Firewall liberado; interface correta
- ☐ Hello MPI OK em todos os nós
- ☐ Execução real do seu script com mpirun/mpiexec