```
import zmq
import msgpack
import time
import asyncio
import requests
import json
from concurrent.futures import ThreadPoolExecutor



class ZeroMQComm:
    def __init__(self, address="tcp://127.0.0.1:5555", io_threads=1):
        self.context = zmq.Context(io_threads=io_threads)
        self.sender = self.context.socket(zmq.PUSH)
        self.sender.bind(address)
        self.receiver = self.context.socket(zmq.PULL)
        self.receiver.connect(address)
        self.executor = ThreadPoolExecutor(max_workers=io_threads)

    async def send(self, msg):
        packed_msg = msgpack.packb(msg)
        await asyncio.get_event_loop().run_in_executor(
            self.executor, self.sender.send, packed_msg
        )

    async def receive(self):
        packed_msg = await asyncio.get_event_loop().run_in_executor(
```

```
            self.executor, self.receiver.recv
        )
        return msgpack.unpackb(packed_msg)

    async def send_batch(self, msgs):
        for msg in msgs:
            await self.send(msg)

    async def receive_batch(self, batch_size):
        messages = []
        for _ in range(batch_size):
            messages.append(await self.receive())
        return messages


# Benchmarking function
async def benchmark():
    # Data to send
    data = {"message": "Hello, World!"}

    # ZeroMQ benchmark
    zmq_comm = ZeroMQComm()
    start_time = time.time()
    await zmq_comm.send(data)
    received_data_zmq = await zmq_comm.receive()
    zmq_duration = time.time() - start_time
```

```
    print(f"ZeroMQ Duration: {zmq_duration:.6f} seconds")

    # HTTP API benchmark
    url = "http://127.0.0.1:5000/send"
    headers = {"Content-Type": "application/json"}

    start_time = time.time()
    response = requests.post(url, data=json.dumps(data), headers=headers)
    received_data_http = response.json()
    http_duration = time.time() - start_time
    print(f"HTTP API Duration: {http_duration:.6f} seconds")

    # Print results
    print(f"ZeroMQ received data: {received_data_zmq}")
    print(f"HTTP API received data: {received_data_http}")


if __name__ == "__main__":
    asyncio.run(benchmark())
```