Submitting a Batch Job

Use `sbatch` to submit a script for batch processing. Your script runs when resources become available — you don't need to stay connected.

Basic Job Script

```
#!/bin/bash
#SBATCH --job-name=my_job                          # Job name
#SBATCH --account=public-users_v2                  # Account name
#SBATCH --partition=power-general-shared-pool      # Partition name
#SBATCH --qos=public                               # QOS type
#SBATCH --time=02:00:00                            # Max run time (hh:mm:ss)
#SBATCH --ntasks=1                                 # Number of tasks
#SBATCH --nodes=1                                  # Number of nodes
#SBATCH --cpus-per-task=10                         # CPU cores required
#SBATCH --mem-per-cpu=4G                           # Memory per CPU
#SBATCH --output=my_job_%j.out                     # Output file (%j = job ID)
#SBATCH --error=my_job_%j.err                      # Error file
#SBATCH --mail-user=your@email.tau.ac.il           # Email address
#SBATCH --mail-type=END,FAIL                       # Notify on completion or failure

module purge
module load gcc/rocky9-gcc-fortran-14.2.0

# Your commands here
./my_program
```

Submit it:

```
sbatch my_job.sh
```

*The maximum time an sbatch job can run is 7 days #SBATCH --time=7-00:00:00

Job Arrays

If you need to submit many similar jobs, use a job array instead of separate `sbatch` calls. Arrays reduce scheduler load and are easier to manage.

```
#!/bin/bash
#SBATCH --job-name=array_job
#SBATCH --account=public-users_v2
#SBATCH --partition=power-general-shared-pool
#SBATCH --qos=public
#SBATCH --time=02:00:00
#SBATCH --ntasks=1
#SBATCH --nodes=1
#SBATCH --cpus-per-task=1
#SBATCH --mem-per-cpu=4G
#SBATCH --array=1-100                          # 100 tasks
#SBATCH --output=array_job_%A_%a.out           # %A = job ID, %a = task ID
#SBATCH --error=array_job_%A_%a.err

# Use SLURM_ARRAY_TASK_ID to differentiate tasks
./my_program input_${SLURM_ARRAY_TASK_ID}.txt
```

Each task runs independently with a different `SLURM_ARRAY_TASK_ID`. Output and error logs are separated per task.

Excluding Nodes

To exclude specific nodes from your job:

```
#SBATCH --exclude=compute-0-[100-103],compute-0-67
```

Chain Jobs

Use `--depend` to run a job only after another completes:

```
sbatch --depend=afterok:45001 next_step.sh
```

Created 2026-06-14 08:01:59 UTC by levk
Updated 2026-07-15 07:25:45 UTC by shoshanaa