按分析需求选择轨迹输出、压缩与保留策略

完整操作教程 · 参考固定 · 往返恢复验收

轨迹太大先问后续要分析什么：结构构象、扩散、速度相关、受力还是快速动力学，需要的信息不同。压缩坐标、减少帧和只保留部分原子都会改变可分析范围。本篇用GROMACS建立输出规划，并从原轨迹另存一个全体系降采样副本，保留原件完成验收。

运行边界：示例代码需按安装版本和真实项目输入复核；未提供真实生物数据，语法或小样本检查不代表科学结果和性能验收。

轨迹太大先问后续要分析什么：结构构象、扩散、速度相关、受力还是快速动力学，需要的信息不同。压缩坐标、减少帧和只保留部分原子都会改变可分析范围。本篇用GROMACS建立输出规划，并从原轨迹另存一个全体系降采样副本，保留原件完成验收。

先区分减少什么信息

XTC主要存压缩坐标，不能替代需要速度或力的TRR等输出；压缩精度也不是无损承诺。只留下蛋白原子后，许多溶剂、离子及接触分析无法补做。增加保存间隔会失去快时间尺度信息，之后插值不能恢复真实运动。输出选择应先与分析目标对应，不以文件越小越好。

选择	可节省内容	需要保留的条件
未来降低写出频率	减少帧数	采样间隔满足目标分析
调整XTC精度	改变坐标量化	先测试目标指标的敏感性
只输出原子子集	减少每帧原子	保存对应拓扑/索引和原子映射
归档checkpoint与输入	保存续算/复查材料	稀疏坐标不是完整checkpoint

把未来输出写入模拟协议

例如时间步长0.002 ps、nstxout-compressed=5000，对应每10 ps一个压缩坐标帧，这只是演算示例。nstxout、nstvout、nstfout决定其他坐标、速度和力输出，不能在不知道后续需求时一律设零。compressed-x-grps和compressed-x-precision应在该版本帮助及最终processed.mdp中确认。

改变输出频率通常需要生成匹配的新TPR并按受控重启流程处理，不能直接修改运行中的mdp后假定立即生效。保留原mdp、processed.mdp、TPR、拓扑和版本，让后续知道每段轨迹保存了什么。接续段的时间轴与输出间隔也应记录。

用实测小段作容量演算

以下保存为estimate_storage.py，四个参数依次为代表小段字节数、小段帧数、计划时长ns、目标间隔ps。例如python3 estimate_storage.py 8000000 100 200 10中的数值均为教学假设，不是实测压缩率。结果包含粗估与20%示例预留，真实工作盘还需计原件、中间文件和备份。

```
import sys,math,json
pilot_bytes,pilot_frames,total_ns,stride_ps=map(float,sys.argv[1:])
assert all(math.isfinite(x) and x>0 for x in [pilot_bytes,pilot_frames,total_ns,stride_
ps])
assert pilot_frames.is_integer() and pilot_bytes.is_integer()
frames=math.floor(total_ns*1000/stride_ps)+1
estimate=pilot_bytes/pilot_frames*frames
result={"planned_frames":frames,"estimated_bytes":estimate,
        "example_20pct_reserve_bytes":estimate*1.2,
        "note":"[redacted]"}
with open("trajectory-storage-plan.json","x") as f:json.dump(result,f,indent=2)
print(result)
```

从现有轨迹另存降采样副本

原轨迹与TPR原子数、顺序和来源已匹配，先用gmx check读取全文件。下面保留System原子，避免本例引入子集拓扑问题；DT_PS填正的目标间隔，单位ps，不能把100当成100帧。-dt按输入时间取帧，需要先检查时间轴精度与连续性，不用改时间戳来掩盖异常。

```
set -euo pipefail
: "${TRAJ:?[redacted]}"
: "${TPR:?[redacted]}"
: "${DT_PS:?[redacted]}"
[[ "$DT_PS" =~ ^[0-9]+([.][0-9]+)?$ ]]
awk -v v="$DT_PS" 'BEGIN{exit !(v>0)}'
test -s "$TRAJ"
test -s "$TPR"
mkdir trajectory-thin-001
sha256sum "$TRAJ" "$TPR" > trajectory-thin-001/inputs.sha256
gmx check -f "$TRAJ" > trajectory-thin-001/original-check.txt 2>&1
printf 'System\n' | gmx trjconv -f "$TRAJ" -s "$TPR" \
  -o trajectory-thin-001/system-thin.xtc -dt "$DT_PS" -ndec 3 \
  > trajectory-thin-001/convert.log 2>&1
test -s trajectory-thin-001/system-thin.xtc
gmx check -f trajectory-thin-001/system-thin.xtc \
  > trajectory-thin-001/output-check.txt 2>&1
```

验收后再决定归档层次

核对输出原子数、起止时间、实际帧数及间隔，抽查对应帧的坐标与计划精度，并用同一分析脚本比较代表指标。-ndec 3控制输出XTC小数精度，不保证原来更高精度信息仍在；原子子集另做时还需保存索引和兼容拓扑，不能把全体系TPR直接当已重排子集的正确配对。

不要在转换命令后自动删除原轨迹。先确定哪些原始资料必须保留、哪些派生轨迹用于日常分析，以及checkpoint和必要输入的保存位置。XTC再次用普通压缩工具压缩未必省多少空间，应按实际小段测量。容量估计持续用真实帧和体积更新，转换期间至少留出原件与新文件同时存在的空间。

配套练习包

本PDF对应的ZIP包含本文代码块、主题练习模板或合成数据、README和SHA256校验文件。请从同一教程页面下载练习包；先核对校验值，再在独立目录复现。

参考资料

GROMACS轨迹输出参数 · 来源资料

gmx trjconv降采样与精度 · 来源资料

gmx check读取轨迹 · 来源资料

操作建议：先在合成小样本上验证流程，再依据项目参考、标签合同、存储策略与获配资源验收真实数据。