ITEM #234 - Hybrid Early-Stop Acceleration for UnalignedAND-BTP (Large-vs-Large)

4A + 4B Head Capture → 2B Composition Top-K → Optional Symmetry Pass

Conversation：UnalignedAND-BTP 算法加速

20260113

Authors: Sizhe Tan & GPT-Obot

DBM-COT ITEM #234 (EN)

Hybrid Early-Stop Acceleration for UnalignedAND-BTP (Large-vs-Large)

4A + 4B Head Capture → 2B Composition Top-K → Optional Symmetry Pass

Category

DBM-COT · Metric Space Intelligence · UnalignedAND-BTP · RTB Acceleration · Minimal Evolution Threshold (MET)

Status

Algorithmic workflow + guardrails (implementation-ready)

---

0. Motivation: Why a Hybrid "Early-Stop" Route

Large-vs-Large UnalignedAND-BTP is an **optimization problem with a long tail**.
Pure "zoom-in forever" acceleration tends to chase diminishing returns: more candidates, more partial states, more micro-optimizations—often violating DBM's **Minimal Evolution Threshold (MET)** principle.

This item proposes a **Hybrid Early-Stop route**:

1. Capture **high-yield head matches** quickly (Step 1–2).
2. Convert the remaining hard search into a **bounded composition problem** (Step 3).
3. Stop when confidence is sufficient, optionally run a **symmetry pass** for extra coverage (Step 4).

The design goal is not to “squeeze every long-tail drop”, but to **stop safely and early** with measurable confidence.

---

1. Problem Definition

Given two large starmaps (complex metric-space points):

- `LargeA`, `LargeB` (GraphStarmap / SequenceStarmap / ImageStarmap / DNB-like structural starmaps)

Goal:

- Output **Top-K global matches** between `LargeA` and `LargeB`, with:
  - non-conflicting mapping/composition,
  - explainable evidence chain,
  - early-stop capability with explicit guardrails.

---

2. Hybrid Algorithm Overview

Step 1 — 4A Head Capture (Full Coverage Pieces on B)

- Cut `LargeA` into blocks `{A_j}` (block semantics preserve core+halo in real impl).
- For each block `A_j`, run **4A (full coverage segmentation)** to find small-to-large matches against `LargeB`.
- Collect matches into a candidate pool `C_4A`.
- Repeat only if necessary (controlled by Stop Rules).

Step 2 — 4B Head Capture (Anchor → Occurrence → CCC Gate → Local Pieces)

- Cut `LargeA` into blocks `{A_j}` (same or updated segmentation).
- For each block `A_j`, run **4B (anchor search + CCC gating)** against `LargeB`.
- Collect matches into a candidate pool `C_4B`.
- Repeat only if necessary (controlled by Stop Rules).

Step 3 — Merge + 2B Composition Top-K (Non-conflicting Combinations)

- Merge: `C = dedup(C_4A ∪ C_4B)` (canonical match key).
- Treat `C` as a set of **Known Matched SubPatterns** in `LargeB` that partially explain `LargeA`.
- Run **2B Composition Solver**:
  - find Top-K **non-conflicting combinations** of subpatterns that best cover/explain `LargeA`,
  - subject to conflict constraints and a residual penalty/lower-bound for uncovered parts.
- Output Top-K composite matches as the global result candidates.

Step 4 — Optional Symmetry Pass (Swap A/B, Re-run, Merge)

To improve coverage and confidence (especially under asymmetry of structural density):

- Swap `LargeA ↔ LargeB`, re-run Steps 1–3.
- Merge the two Top-K sets, dedup, and optionally re-score by mutual consistency.

---

3. Why This is "Early-Stop" and MET-Friendly

This hybrid route shifts the long tail from "deep matching search" into a bounded composition space:

- 4A and 4B act as **head-yield subpattern harvesters**.
- 2B composition acts as **structured assembly**, typically far smaller than the raw BTP state space.
- Stop Rules define when to quit collecting more subpatterns and commit to composition.

---

4. Mermaid: End-to-End Map

```
flowchart TD
  A[Input: LargeA, LargeB] --> B[Cut LargeA into blocks {A_j}]
  B --> C1[Step 1: 4A Head Capture\nA_j vs LargeB (full-coverage pieces)]
  B --> C2[Step 2: 4B Head Capture\nAnchors->Occurrences->CCC Gate->local
pieces]
  C1 --> D[Merge + Dedup Candidates\nC = C_4A ∪ C_4B]
  C2 --> D
  D --> E[Step 3: 2B Composition Solver\nTop-K non-conflicting
combinations\n+ residual penalty / LB]
  E --> F[Top-K Global Matches (Hybrid)]
  F --> G{Optional Symmetry Pass?}
  G -- Yes --> H[Swap A/B, rerun Steps 1-3]
```

```
H --> I[Merge two Top-K sets\nDedup + consistency rescore]
G -- No --> J[Done]
I --> J[Done]
```

---

5. Contracts and Stop Rules

5.1 Contract Table

Contract	Purpose	Exactness Guard (EXACT Mode)
C1 PieceWithHalo	prevent boundary silent misses	halo width ≥ Reach(A_j), core coverage across domain
C2 AnchorSet	candidate recall control	anchors must not exclude unless must-hit is provable
C3 OccurrenceCccGate	filter noisy occurrences	cannot REJECT by quality-only; only by LB evidence; else DEFER
C4 MatchKey/Conflict	deterministic dedup + conflict constraints	canonical key; conflicts explicit; no overlapping/inconsistent combos
C5 Composition 2B	assemble candidates into global matches	must respect conflict graph; objective includes residual LB/penalty

5.2 Stop-Rule Table (Minimal Evolution Threshold “Brake System”)

Stop Rule	Trigger	Action	MET Rationale
S1 Marginal Gain	coverage gain < ε_cov OR top-K improvement < ε_cost	stop collecting more candidates	long-tail returns too small
S2 LB Sealing	best_combo_cost + LB_remain ≥ kth_combo_cost	stop; Top-K stable	provable “no further improvement”
S3 Candidate Explosion	#candidates or #conflictEdges exceeds cap	tighten gate / anchor rarity OR jump to Step 3	avoid drowning in tail noise

---

6. Output Semantics (What “Top-K” Means Here)

Hybrid Top-K is **exact** if one of the following holds:

- (P1) Candidate completeness: the optimal solution’s necessary subpatterns are present in `C`; or
- (P2) Lower-bound sealing: `LB_remain` proves no unseen structure can beat current Top-K.

Otherwise, Hybrid Top-K is a **high-confidence bounded approximation**, with explicit metrics:

- coverage %, residual mass, ambiguity, symmetry-consistency score.

---

7. Practical Guidance

Recommended defaults for 2026 engineering:

- Use Hybrid as the default **Large-vs-Large** driver.
- Start with EXACT-safe gating (DEFER heavy, REJECT only by LB) until stabilized.
- Enable symmetry pass once (ONCE) when coverage is uncertain.
- Keep stop rules enabled from day one to prevent “never-ending zoom-in”.

DBM-COT ITEM #234 (中文)

UnalignedAND-BTP（大对大）Hybrid 及时收手加速路线

4A + 4B 头部捕获 → 2B 组合 Top-K → 可选对称增信

分类

DBM-COT · 度量空间智能 · UnalignedAND-BTP · RTB 加速 · 最小进化门槛（MET）

状态

算法流程 + 护栏（可直接工程落地并迭代硬化）

---

0. 动机：为什么要 Hybrid + 及时收手

大对大 UnalignedAND-BTP 属于典型的**长尾优化问题**：
如果不断 zoom-in、不断加候选、不断局部抠细节，很容易陷入“低收益长尾”——不记工本地追逐边角收益，这违背 DBM 的 **最小进化门槛（MET）** 理念。

本 ITEM 提出一条 **Hybrid 及时收手路线**：

1. 先快速抓住**高收益头部匹配块**（Step 1–2）
2. 把剩余难题降维成一个**规模可控的组合拼装问题**（Step 3）
3. 达到足够置信度后及时收手；如需增信，再做一次**对称互换**（Step 4）

目标不是“榨干每一滴长尾”，而是**在可控时间内得到可解释、可增信、可早停的 Top-K**。

---

1. 问题定义

输入两张大型 starmap（复杂 metric-space 点）：

- `LargeA`, `LargeB`（图/序列/图像/DNB 类结构 starmap）

输出：

- `LargeA` 与 `LargeB` 的 **Top-K 全局匹配**（可解释、无冲突组合），并具备明确的早停机制与护栏。

---

2. Hybrid 算法全流程

Step 1 — 4A 头部捕获（全覆盖切片）

- 将 `LargeA` 切成块 `{A_j}`（真实实现应保留 core+halo 语义）。
- 对每个 `A_j`，用 **4A（B 全域切片）** 在 `LargeB` 中找 small-to-large matches。
- 收集候选集合 `C_4A`。
- 是否重复，受 Stop Rules 控制（避免长尾）。

Step 2 — 4B 头部捕获（Anchor→Occurrence→CCC Gate→局部切片）

- 将 `LargeA` 切成块 `{A_j}`（可复用或更新切法）。
- 对每个 `A_j`，用 **4B（锚定检索 + CCC gate）** 在 `LargeB` 中找 matches。
- 收集候选集合 `C_4B`。
- 是否重复，受 Stop Rules 控制（避免长尾）。

Step 3 — 合并 + 2B 组合 Top-K（无冲突组合）

- 合并：`C = dedup(C_4A ∪ C_4B)`（必须 canonical match key 去重）。
- 把 `C` 看作 `LargeB` 中一组 **已知匹配子模式（Known Matched SubPatterns）**，它们可部分解释 `LargeA`。
- 运行 **2B 组合求解器**：
  - 从子模式中选择 **无冲突组合**，尽可能覆盖/解释 `LargeA`；
  - 目标函数 = 组合成本（距离） + 未覆盖残差的惩罚或下界（LB）。
- 输出 Top-K 组合解释，作为 Hybrid 的全局匹配候选。

Step 4 — 可选对称增信（互换 A/B 再跑一轮）

为提高覆盖与置信（尤其在结构密度不均、锚点稀有度不均时）：

- 互换 `LargeA ↔ LargeB`，再跑 Step 1–3；
- 合并两边 Top-K，去重，并可用双向一致性重打分。

---

3. 为什么它是“及时收手”且符合 MET

这条路线把长尾从“深度匹配空间”搬到“可控组合空间”：

- 4A/4B 是“头部高收益块捕获器”
- 2B 是“结构拼装器”，规模通常远小于原始 BTP 搜索空间
- Stop Rules 是“鱼控刹车系统”，防止无限 zoom-in

---

4. Mermaid 总图

```
flowchart TD
  A[输入: LargeA, LargeB] --> B[切 LargeA 为块 {A_j}]
  B --> C1[Step 1: 4A 头部捕获\nA_j vs LargeB (全覆盖切片)]
  B --> C2[Step 2: 4B 头部捕获\n 锚定->occ->CCC Gate->局部切片]
  C1 --> D[合并+去重候选\nC = C_4A ∪ C_4B]
  C2 --> D
  D --> E[Step 3: 2B 组合求解器\nTop-K 无冲突组合\n+ 残差惩罚/LB]
  E --> F[Hybrid Top-K 全局匹配]
  F --> G{可选对称增信?}
  G -- 是 --> H[互换 A/B 重跑 Step 1-3]
  H --> I[合并两边 Top-K\n 去重+一致性重打分]
  G -- 否 --> J[结束]
  I --> J[结束]
```

---

5. Contract / Stop-Rule 表

5.1 Contract 表（护栏）

Contract	作用	EXACT 模式护栏要点
C1 PieceWithHalo	防边界静默漏解	halo ≥ Reach(A_j)，core 覆盖域可控
C2 AnchorSet	控制召回与盲区	anchors 不得排除候选，除非 must-hit 可证明
C3 OccurrenceCccGate	删噪声 occurrence	不得仅凭质量 REJECT；只能用 LB 证据 REJECT，否则 DEFER
C4 MatchKey/Conflict	去重 + 冲突约束	key 必须 canonical；冲突显式化；组合不得重叠/不一致
C5 2B 组合器	全局拼装求解	必须尊重冲突图；目标含残差惩罚/LB

5.2 Stop Rules（MET “刹车系统”）

Stop Rule	触发条件	动作	MET 含义
S1 边际收益阈值	覆盖增量 < ε_cov 或 top-K 改善 < ε_cost	停止继续采集候选	长尾收益太小应收手
S2 下界封顶	best_combo_cost + LB_remain ≥ kth_combo_cost	停止，Top-K 已稳定	可证明继续无意义
S3 候选爆炸熔断	候选数/冲突边数超过上限	收紧 gate/锚点稀有度或直接进入 Step 3	防止长尾噪声淹没系统

---

6. 输出语义：何时是“严格 Top-K”，何时是“高置信近似”

Hybrid 输出为严格 Top-K 的条件之一：

- (P1) 候选完备：最优解所需子块都在 `C` 中；或
- (P2) 下界封顶：`LB_remain` 证明未捕获部分不可能改写 Top-K。

否则它是**高置信、有指标的近似**，并应输出：

- 覆盖率、残差质量、歧义度、对称一致性分数等。

---

7. 工程建议（2026 默认姿态）

- Hybrid 作为 **大对大默认驱动器**；
- 早期以 EXACT-safe gate 为主（DEFER 多，REJECT 只靠 LB）；
- 对称 pass 默认 ONCE（一次增信），避免无止境；
- Stop Rules 从第一天就启用，防止系统掉入长尾深坑。
