

GPU-Accelerated Collocation Pattern Discovery

Térbeli együttes előfordulási minták GPU-val gyorsított
felismerése

Gyenes Csilla
Sallai Levente
Szabó Andrea

Eötvös Loránd Tudományegyetem
Informatikai Kar

2013. november 26.

Tartalom

- 1 Bevezetés
 - Motiváció
 - Alapfogalmak
 - Megvalósítás
- 2 iCPI-fa
- 3 GPU-CM
 - CUDA
 - Eredmények
- 4 Összefoglalás

Tartalom

- 1 **Bevezetés**
 - Motiváció
 - Alapfogalmak
 - Megvalósítás
- 2 iCPI-fa
- 3 GPU-CM
 - CUDA
 - Eredmények
- 4 Összefoglalás

Feladat

GPU-Accelerated Collocation Pattern Discovery

Witold Andrzejewski, Pawel Boinski

Poznan University of Technology, Institute of Computing
Science, Piotrowo 2, 60-965

Poznan, Poland

Motiváció

Példák:

Motiváció

Példák:

- Fajok, üzleti típusok POI-k (kórházak, repterek, látnivalók)

Motiváció

Példák:

- Fajok, üzleti típusok POI-k (kórházak, repterek, látnivalók)
- Viselkedésanalízis: mobiltelefon hálózatot üzemeltető cég vizsgálhatja a kapcsolatot az ügyfelek és bizonyos szolgáltatások választása között

Motiváció

Példák:

- Fajok, üzleti típusok POI-k (kórházak, repterek, látnivalók)
- Viselkedésanalízis: mobiltelefon hálózatot üzemeltető cég vizsgálhatja a kapcsolatot az ügyfelek és bizonyos szolgáltatások választása között
- Továbbá ökológiában, meteorológiában az együttesen előforduló természeti jelenségek vizsgálatakor

Alapfogalmak

Egybefüggő minták

Olyan térbeli objektumok, amelyek gyakran együtt, egymás térbeli szomszédaiként helyezkednek el.

Alapfogalmak

Egybefüggő minták

Olyan térbeli objektumok, amelyek gyakran együtt, egymás térbeli szomszédaiként helyezkednek el.

Térbeli objektumok

A térbeli objektumok definiálhatóak a tér egy adott pontbeli tulajdonságával.

Alapfogalmak

Részvételi arány

Az f_i tulajdonság részvételi arányát a C egybefüggő mintára vonatkoztatva $Pr(C, f_i)$, $C = \{f_1, \dots, f_k\}$ jelöli.

$Pr(C, f_i)$ az f_i C -beli példányainak száma osztva az összes f_i tulajdonságú elemmel.

Alapfogalmak

Részvételi arány

Az f_i tulajdonság részvételi arányát a C egybefüggő mintára vonatkoztatva $Pr(C, f_i)$, $C = \{f_1, \dots, f_k\}$ jelöli.

$Pr(C, f_i)$ az f_i C -beli példányainak száma osztva az összes f_i tulajdonságú elemmel.

Részvételi index

A részvételi indexe $P_i(C)$, egy $C = \{f_1, \dots, f_k\}$ csoportnak:

$$P_i(C) = \min_{f_i \in C} Pr(C, f_i).$$

Megvalósítás

- Korábban az egybefüggő mintákat felismerő algoritmusok implementálása CPU-n történt

Megvalósítás

- Korábban az egybefüggő mintákat felismerő algoritmusok implementálása CPU-n történt
- Újítás a hardver támogatás kihasználása, azaz a minták a grafikus kártyán, a GPU-n történő felismerése

Megvalósítás

- Korábban az egybefüggő mintákat felismerő algoritmusok implementálása CPU-n történt
- Újítás a hardver támogatás kihasználása, azaz a minták a grafikus kártyán, a GPU-n történő felismerése
- Erre módszer:
GPU-CM algoritmus, ami az iCPI-fa GPU-val gyorsított verziója

Tartalom

- 1 Bevezetés
 - Motiváció
 - Alapfogalmak
 - Megvalósítás
- 2 iCPI-fa
- 3 GPU-CM
 - CUDA
 - Eredmények
- 4 Összefoglalás

iCPI-fa

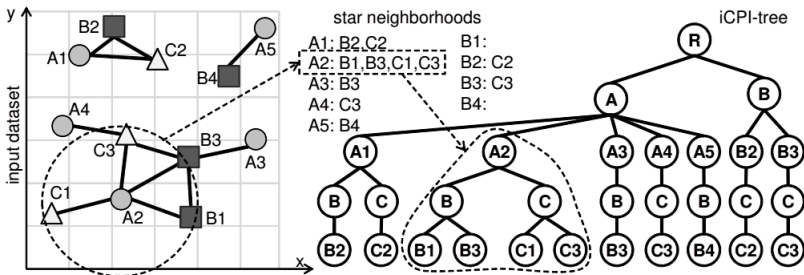


Fig. 1. Sample dataset and the corresponding iCPI-tree

iCPI-fa a különböző növények eloszlására

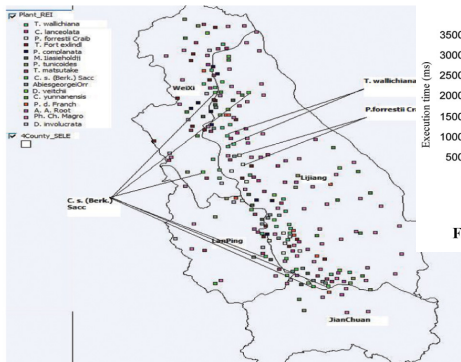


Fig. (11). An example of the distribution of plant data.

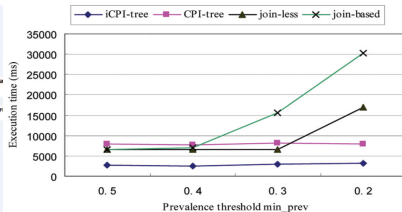


Fig. (9). A comparison using a plant distribution dataset.

Tartalom

- 1 Bevezetés
 - Motiváció
 - Alapfogalmak
 - Megvalósítás
- 2 iCPI-fa
- 3 GPU-CM**
 - CUDA
 - Eredmények
- 4 Összefoglalás

GPU-CM algoritmus

Bemenet: iCPI-fa

GPU-CM algoritmus

Bemenet: iCPI-fa

Kimenet: gyakori egybefüggő minták és gyakoriságuk

GPU-CM algoritmus

Bemenet: iCPI-fa

Kimenet: gyakori egybefüggő minták és gyakoriságuk

(a) az iCPI-fa beolvasása egy hashmapbe

GPU-CM algoritmus

Bemenet: iCPI-fa

Kimenet: gyakori egybefüggő minták és gyakoriságuk

(a) az iCPI-fa beolvasása egy hashmapbe

(b) az egyelemű minták építése a példányokból

GPU-CM algoritmus

Bemenet: iCPI-fa

Kimenet: gyakori egybefüggő minták és gyakoriságuk

- (a) az iCPI-fa beolvasása egy hashmapbe
- (b) az egyelemű minták építése a példányokból
- (c) az eddigi minták kibővítése, összefűzhető minták keresése és összefűzése

GPU-CM algoritmus

Bemenet: iCPI-fa

Kimenet: gyakori egybefüggő minták és gyakoriságuk

- (a) az iCPI-fa beolvasása egy hashmapbe
- (b) az egyelemű minták építése a példányokból
- (c) az eddigi minták kibővítése, összefűzhető minták keresése és összefűzése
- (d) mintajelöltek ritkítása: azon minta jelöltek eldobása, amelyekre a gyakoriság kisebb, mint az adott küszöb

GPU-CM algoritmus

Bemenet: iCPI-fa

Kimenet: gyakori egybefüggő minták és gyakoriságuk

- (a) az iCPI-fa beolvasása egy hashmapbe
- (b) az egyelemű minták építése a példányokból
- (c) az eddigi minták kibővítése, összefűzhető minták keresése és összefűzése
- (d) mintajelöltek ritkítása: azon minta jelöltek eldobása, amelyekre a gyakoriság kisebb, mint az adott küszöb
- (e) mintapéldányok létrehozása

GPU-CM algoritmus

Bemenet: iCPI-fa

Kimenet: gyakori egybefüggő minták és gyakoriságuk

- (a) az iCPI-fa beolvasása egy hashmapbe
- (b) az egyelemű minták építése a példányokból
- (c) az eddigi minták kibővítése, összefűzhető minták keresése és összefűzése
- (d) mintajelöltek ritkítása: azon minta jelöltek eldobása, amelyekre a gyakoriság kisebb, mint az adott küszöb
- (e) mintapéldányok létrehozása
- (f) (c)-től újratekintve végigiterálunk a fán

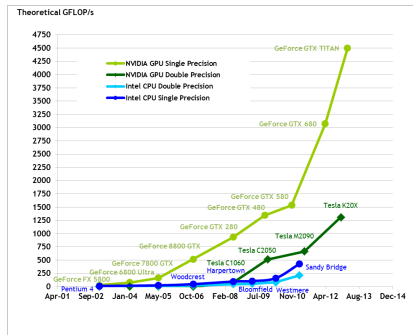
CUDA

Mi a CUDA?

Compute Unified Device Architecture

Az NVIDIA párhuzamos architektúra modellje a saját hardvereihez.

- A programok a kernelek,
- Több példányban futnak, minden példány egy szál,
- A szálak blokkot alkotnak, melyet az SMX dolgoz fel,
- Az SMX 32 szálanként hajtja végre a blokkot



CUDA

CUDA architektúra



Eredmények

Három mérés a CPU-s és GPU-s algoritmus sebességének összehasonlítására:

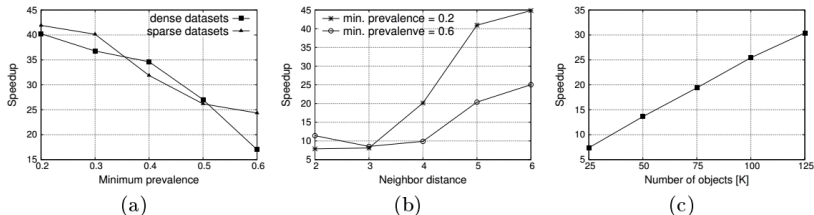


Fig. 2. Experiment results

Tartalom

- 1 Bevezetés
 - Motiváció
 - Alapfogalmak
 - Megvalósítás
- 2 iCPI-fa
- 3 GPU-CM
 - CUDA
 - Eredmények
- 4 **Összefoglalás**

További lehetőségek

- Javaslát

- Probléma: jelenleg a memóriát pazarolja, akkor is tárol értékeket és végez számításokat, ha logikailag üresek az elemek.
- Megoldás: változó hosszú listák kezelése az algoritmusban

- Javaslát

- Probléma: a GPU memóriája korlátozott
- Megoldás: olyan megoldás bevezetése, ami memória korlátos környezetben is működik (partíciós együttálló minta bányászat)

További lehetőségek

- **Javaslat**
 - Probléma: Az iCPI-fa építése CPU-n zajlik
 - Megoldás: hatékony megoldás keresése a GPU-ra való átállásra

Új cél: az algoritmus hangolása, azaz elérni a maximálisan kapható gyorsulást GPU-val.

Vége

Köszönjük a figyelmet!