

1

Vektorisierung: darstellung als Vektoren und in Folge SIMD ->Voraussetzung: Intern keine Dependencies->größere Register helfen hier

„Parallelisierung“ im Kern

Wenn Daten in Cache ca. *4

Wenn zu groß und in Ram ca. *1.5

Bei größerem Abstand kann die dif vom abstand ins register geladen werden

eher wenigerzusätzliche datenstrukturen

Höchstens alignment und vor allem neue instruktionen, Benutzung von Vektorregistern

costmodel analysis (lohnt sich Vektorisierung überhaupt?)

-ausetzen kostet schon (load operationen z.B)

-bei runtime abhängigkeiten kann compiler multiversions für code generieren

Vektorregister werden trd oft mit skalaren operationen verwendet

2

Snippet1

Evtl alias

Compile time vs linktime ->libraryfunctions problematisch ->auch .a oder .so funktionieren nicht.

Restrict dazuschreiben bei x und y->user garantiert, dass beides nicht mehr überlappt

Snippet2

Loop normalisation vor allem wegen step unification

Snippet3

Keine dependencies