ВСЕРОССИЙСКАЯ НАУЧНАЯ КОНФЕРЕНЦИЯ

НАУЧНЫЙ СЕВИС В СЕТИ ИНТЕРНЕТ

Автоматизация распараллеливания программ в системе SAPFOR

Н.А. Катаев[1], С.А. Черных[2]

[1]Институт прикладной математики им. М.В. Келдыша РАН

[2]Московский государственный университет им. М.В. Ломоносова

23 сентября 2020 г. | Москва

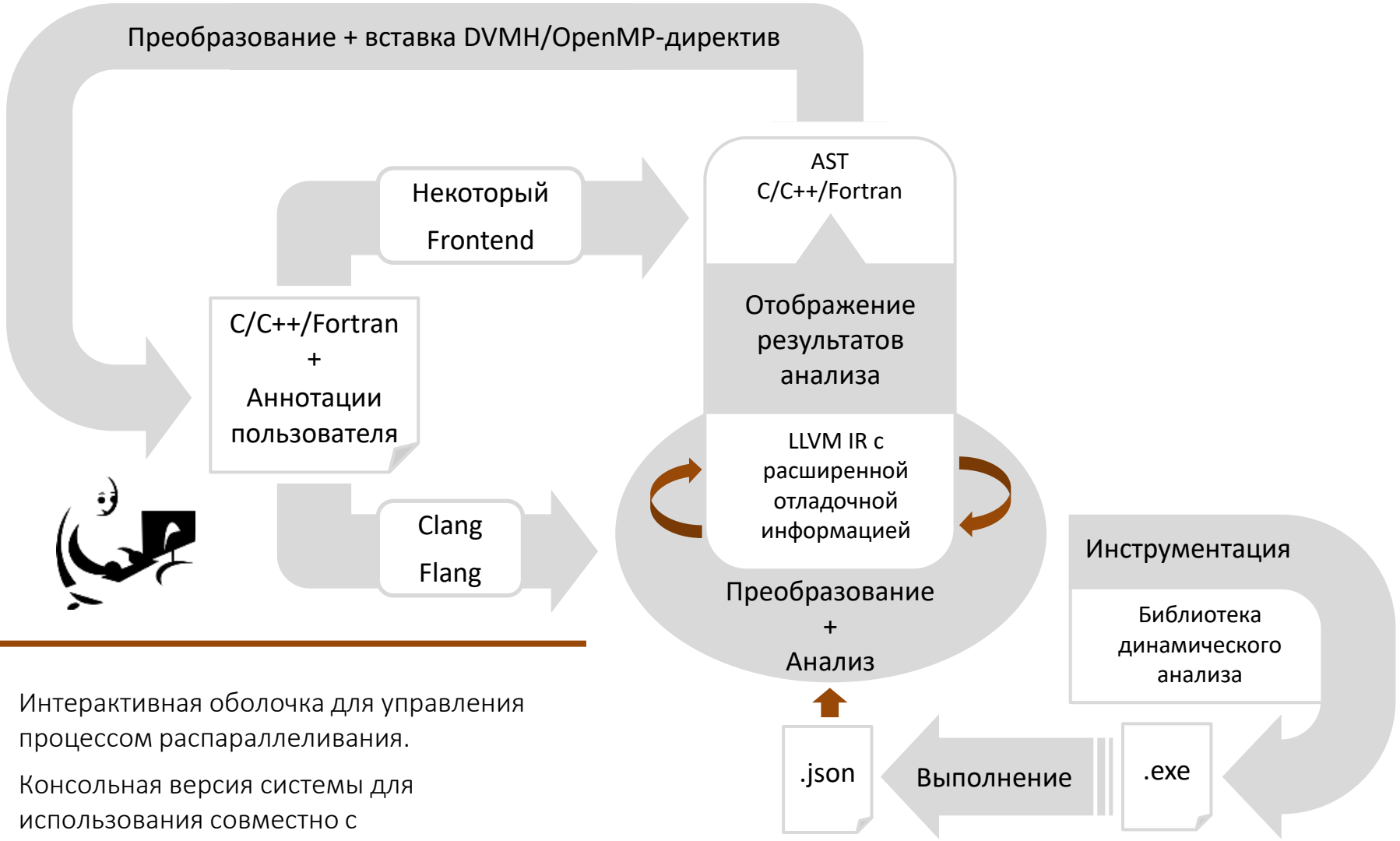
SAPFOR (System FOR Automated Parallelization)

- Исследование характеристик и свойств программы (профилирование, анализ зависимостей по данным и др.).
- Автоматическое распараллеливание «хорошо» написанных потенциально параллельных программ.

  Следование определенным правилам при разработке программ на традиционных языках программирования, дополнительное описание свойств программ.
- Автоматизированное преобразование программ к потенциальному параллельному виду.

  Устранение зависимостей в программе, оптимизация доступа к памяти, изменение структуры хранения данных и структуры вычислений (подстановка процедур и переменных, преобразование циклов и др.).

Внутреннее устройство SAPFOR



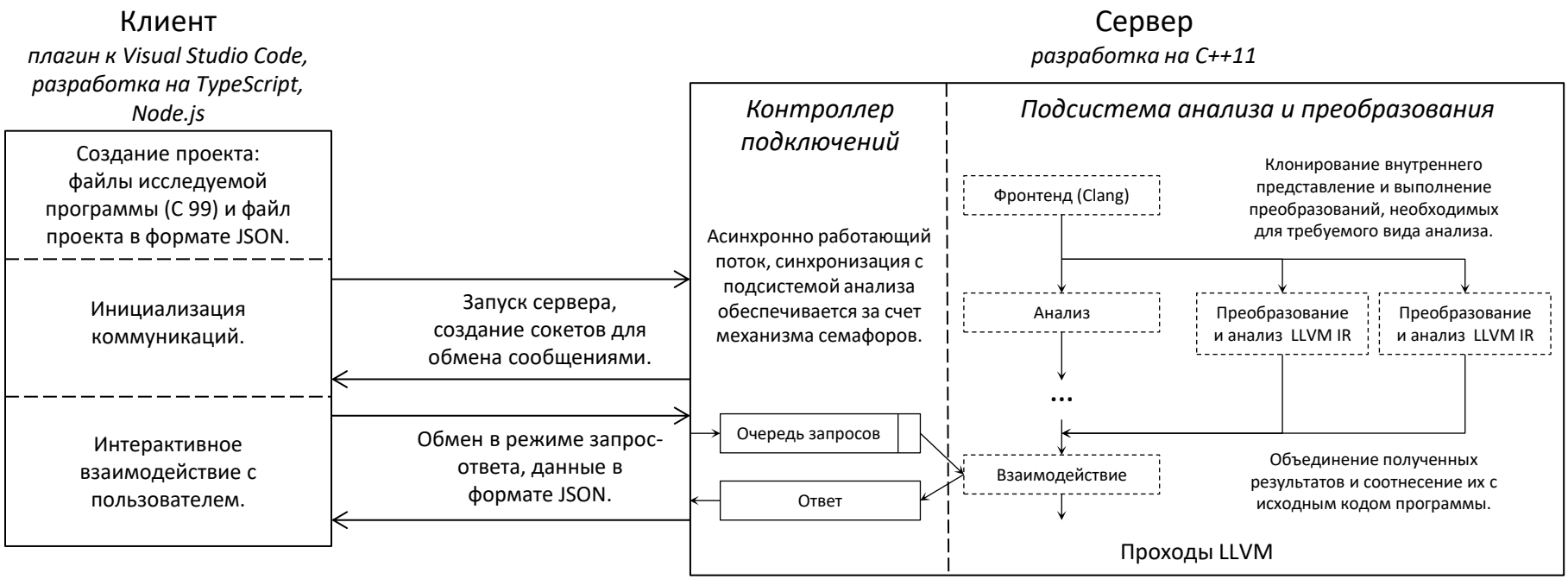
- Интерактивная оболочка для управления процессом распараллеливания.
- Консольная версия системы для использования совместно с автоматизированными средствами сборки.

Процесс распараллеливания в SAPFOR

1. Подготовка программы пользователем к анализу и преобразованию средствами системы SAPFOR, указание опций необходимых для синтаксического разбора программы.
2. Профилирование программы (механизм интервалы DVM системы, сторонние инструменты профилирования) с целью определить наиболее времяемкие участки кода, которые требуется распараллелить в первую очередь.
3. Статические и динамический анализ программы (возможность для пользователя задать некоторые свойства программы вручную, в том числе использую глобальные опции анализа). Пользователь использует интерактивную оболочку, чтобы изучить потенциал параллелизма доступного в программе и обнаруженного SAPFOR.
4. Выполнение преобразований программы, необходимых для распараллеливания фрагментов, выбранных пользователем на основе результатов анализа, профилирования и рекомендаций системы. Пользователь может разметить исходный код и запросить автоматическое выполнение преобразований, либо преобразовать программу вручную.
5. Автоматическое распараллеливание полученной потенциально параллельной программы. Пользователь может разметить области программы, которые требуется распараллелить, либо запустить автоматическое распараллеливание всей программы.

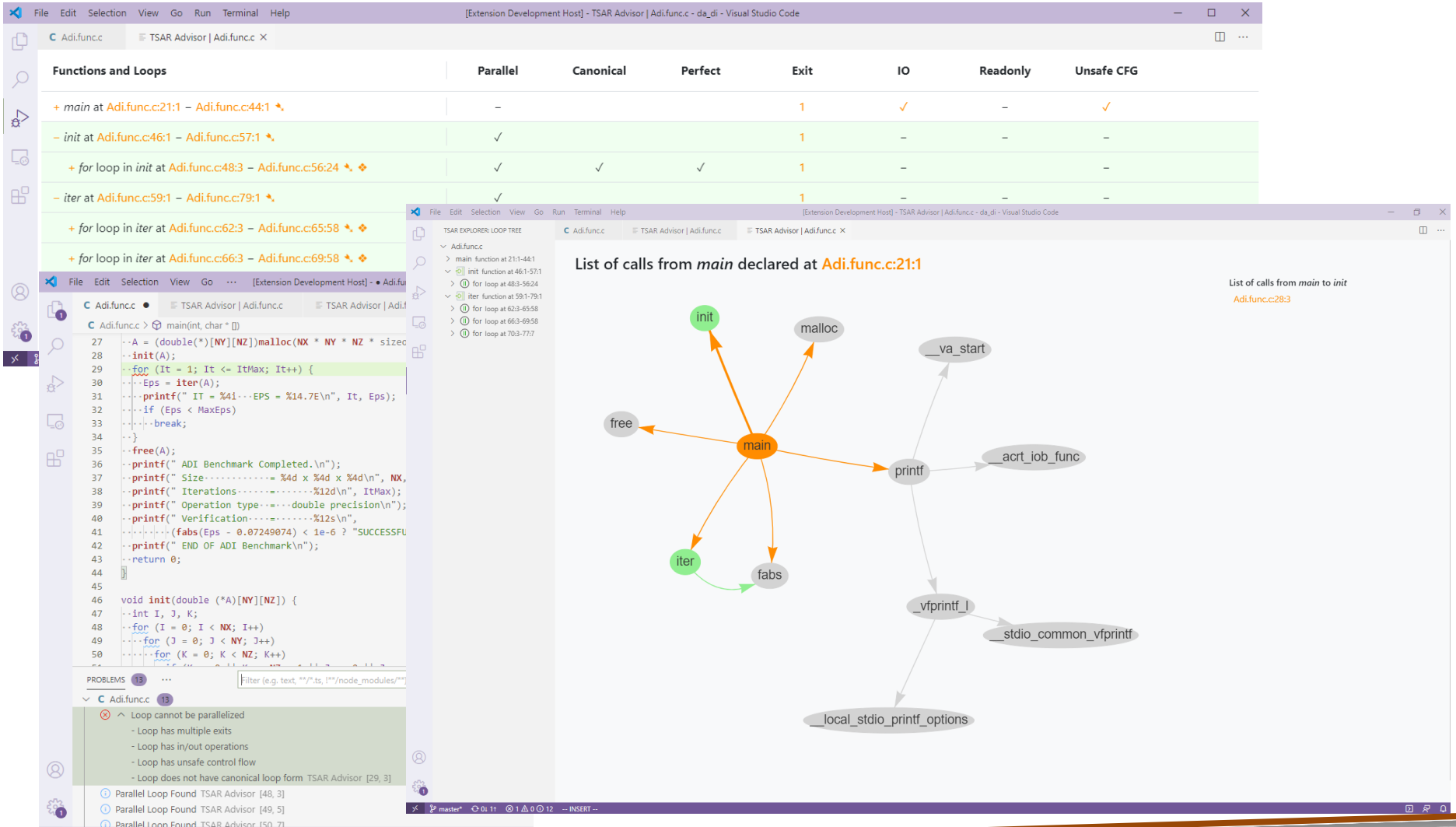
Интерактивная оболочка SAPFOR

- Клиент-серверное взаимодействие между диалоговой оболочкой и ядром системы (средствами анализа, преобразования, принятия решений по распределению данных и вычислений и организации межпроцессорных коммуникаций).
- Серверная часть может быть развернута как локально, на вычислительной системе пользователя, так и на удаленном сервере, доступном через SSH.
- Плагин для кроссплатформенного редактора Microsoft Visual Studio Code.



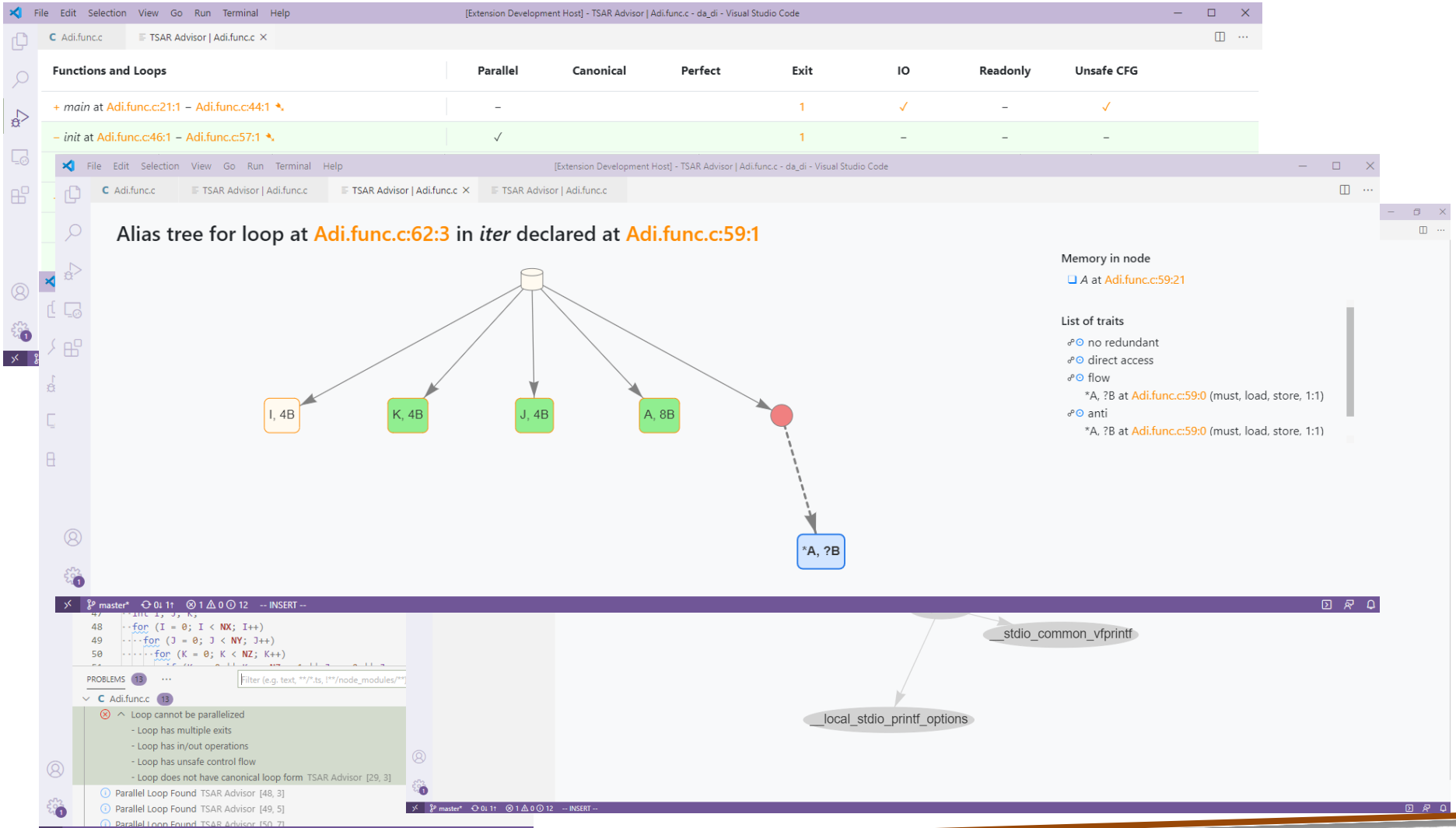
Интерактивная оболочка: общая информация

Для каждого цикла в программе доступна следующая информация: является ли поток управления внутри тела цикла безопасным для распараллеливания, находится ли цикл в канонической форме, присутствуют ли в цикле обращения к памяти, порождающие зависимости по данным.



Интерактивная оболочка: анализ зависимостей

Зависимости по данным между различными итерациями цикла будут классифицированы в соответствии с возможностями их устранения (редукционные и индуктивные переменные, переменные, зависимость по которым может быть устранена за счет приватизации данных или за счет организации конвейерного выполнения цикла).



Автоматическое распараллеливание в модели DVMH

Для параллельного выполнения программ на мультипроцессоре или графическом ускорителе необходимо определить:

- циклы программы, которые могут быть выполнены параллельно.

  Допускается параллельное выполнение циклов for, представленных в канонической форме, для которых может быть гарантирована безопасность потока управление и отсутствие неустранимых зависимостей по данным, возможно использование редукционных переменных и переменных, которые могут быть объявлены приватными.

- данные, которые должны быть переданы на ускоритель и возвращены на процессор после вычислений.

Определить входные, выходные и локальные данные для параллельного цикла.

```
for (…) {
   K = …     // local
   … = B[I]  // in
   A[K] = …  // out
}
```

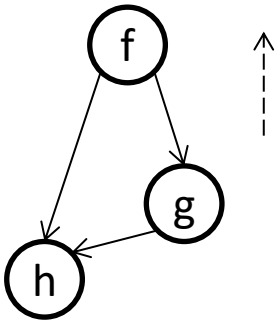
⇨

```
#pragma actual(B)
  for (…) {…}
#pragma get_actual(A)
```

1

Проанализировать граф вызовов в направлении от вызывающих функций к вызываемым для сокращения объема передаваемых данных.

```
h() -> g() -> f()
```



2

Оптимизация обменов данными с ускорителем

Объединение спецификаций актуализации данных соседних циклов.

```
#pragma actual(B)
  for (…) {…}
#pragma get_actual(A)
#pragma actual(B)
  for (…) {…}
#pragma get_actual(A)
```

⇨

```
#pragma actual(B)
  for (…) {…}
  for (…) {…}
#pragma get_actual(A)
```

2.a

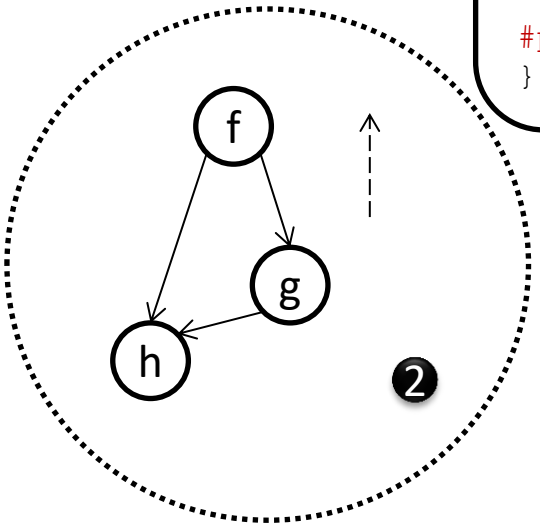
Вынос спецификаций актуализации данных из объемлющих последовательных циклов.

```
for (…) {
#pragma actual(B)
#pragma parallel (1)
  for (…) {…}
#pragma get_actual(A)
}
```

⇨

```
#pragma actual(B)
for (…) {
  for (…) {…}
}
#pragma get_actual(A)
```

2.b



Перенос спецификаций актуализации данных из вызываемой функции в точку вызова.

```
foo();

void foo() {
#pragma actual(B)
  …
#pragma get_actual(A)
}
```

⇨

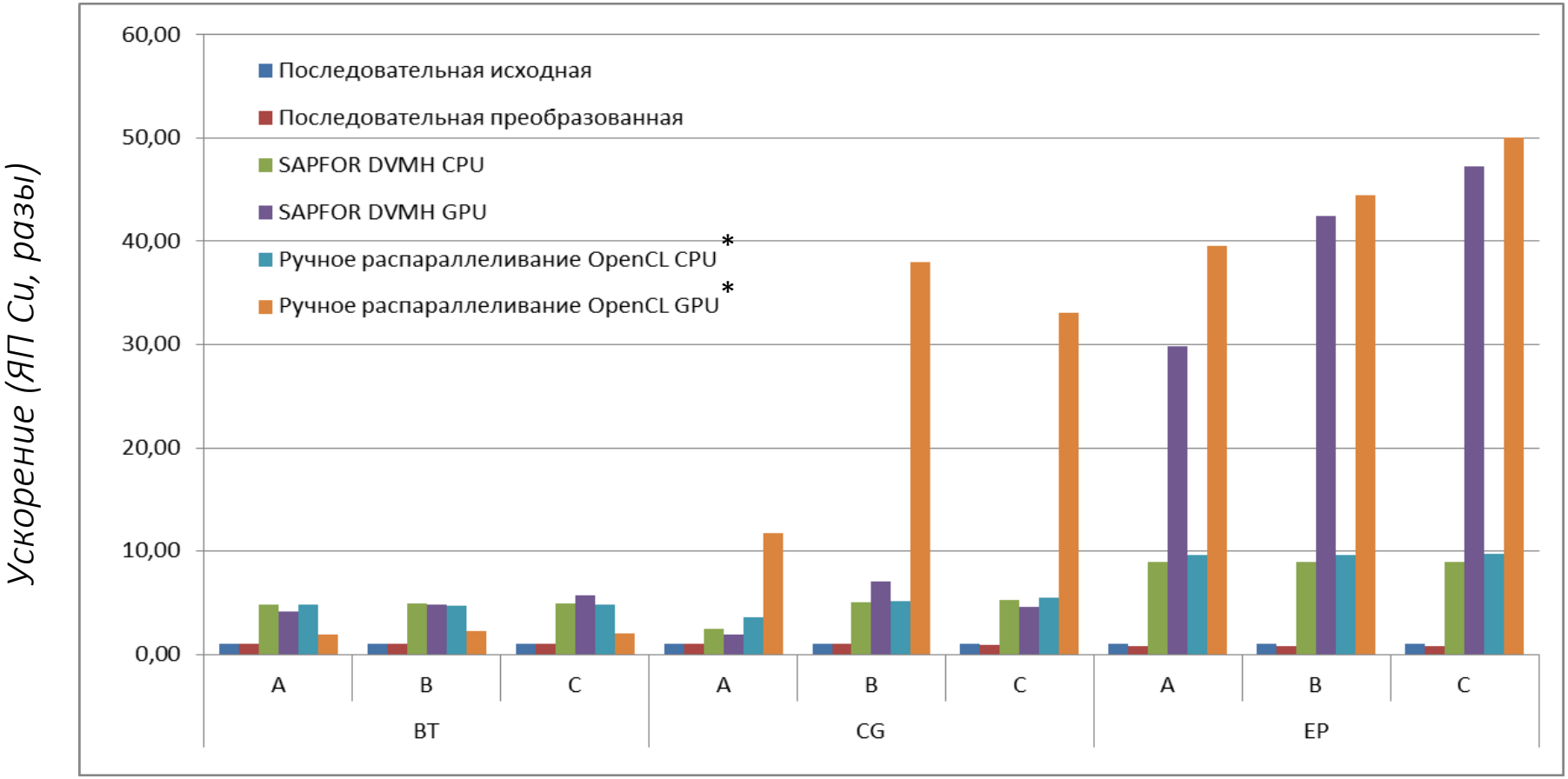
```
#pragma actual(B)
foo();
#pragma get_actual(A)

void foo() {…}
```

2.c

Распараллеливание тестов NAS Parallel Benchmarks 3.3.1

Программы в модели DVMH, остаются написанными в последовательном стиле, а параллелизм в них выражен за счет высокоуровневых спецификаций, в отличие от OpenCL версий программ, которые существенно отличаются от исходных программ.



* Seo S., Jo G., Lee J. Performance Characterization of the NAS Parallel Benchmarks in OpenCL // 2011 IEEE International Symposium on. Workload Characterization (IISWC), 2011. — P. 137-148

GPU GeForce GTX 1660 Ti
CPU Intel Xeon E5-1660 v2, 3.70 GHz, с отключенным Turbo Boost (6 ядер, 12 потоков).
Intel Compiler 19.0.2.187 и NVIDIA Compiler 10.2 с опцией оптимизации –O3

Заключение

Система SAPFOR автоматизирует процесс разработки параллельных программ, позволяя исследовать свойства программ, выполнять автоматическое распараллеливание «хорошо» написанных потенциально параллельных программ, и упрощает приведение программ к потенциально параллельному виду.

Разработанная интерактивная оболочка упрощает участие пользователя в процессе распараллеливания, позволяя ему изучить принимаемые системой решения и при необходимости дать соответствующие рекомендации системе.

Используя интерактивную оболочку пользователь может влиять как на выполнение анализа программы, подсказывая системе наличие тех или иных свойств программы, так и на выполнение преобразований исходного кода, необходимых для распараллеливания программы.

Полученная в итоге программа на последовательном языке программирования может быть автоматически распараллелена системой SAPFOR.

Исходные коды доступны на GitHub: https://github.com/dvm-system

Спасибо за внимание

http://dvm-system.org

https://github.com/dvm-system