並列ファイルシステムのための効率的なシステムコールフックライブラリの設計と評価

情報システム主専攻　202111759　宮内遥楓

指導教員　建部修見

提出日　2024 年 10 月 24 日

1　序論

ユーザー空間並列ファイルシステムは、ストレージシステムの性能を向上させるために開発されてきた [1, 2, 3]。一方、POSIX インターフェースは、標準として長い間アプリケーションに使用されてきた。多くのアプリケーションを動作させるためには POSIX インタフェースのサポートが必要であるが、FUSE やシステムコールインターセプションライブラリなどの既存の手法には様々な問題がある。本研究では、バイナリ書き換えに基づくシステムコールフック機構である zpoline [4] を利用することでこの問題を解決し、その性能結果を示す。

2　既存手法

2.1　FUSE

FUSE（Filesystem in Userspace）は、ユーザ空間ファイルシステムを実装するのに広く用いられている。FUSE はユーザー空間で POSIX インターフェースを完全にサポートできるが、オーバーヘッドが大きい。

2.2　システムコールインターセプトライブラリ

gotcha [5] のようなシステムコールインターセプトライブラリは、libc（C 標準ライブラリ）の関数呼び出しをインターセプトする。libc(C 標準ライブラリ) の関数と同じ名前の関数を定義し、LD_PRELOAD を使うことで、システムコールインターセプトライブラリで定義した関数が呼び出される。しかしながら、libc の関数はシステムコールに対応していないので全てのシステムコールをフックすることができない、libc 内でマクロを利用して呼び出されたシステムコールをフックできないなどの問題がある。

3　設計

本研究では、並列ファイルシステムの 1 つである CHFS を対象にシステムコールフックライブラリを設計・実装する。提案するシステムコールフックライブラリは、zpoline を用い、read、write、open、close、stat、lstat、lseek、pread64、pwrite64、openat、fsync、newfstat システムコールをフックする。ファイルにアクセスした際、パス名が仮想マウントポイント/chfs の下にある場合、CHFS のクライアントライブラリを呼び出し、そうでない場合は元のシステムコールを呼び出す。

4　評価

本研究では、zpoline を用いたシステムコールフックライブラリの性能を、CHFS の API を直接呼び出した場合や FUSE を使用した場合と比較して評価した。Pegasus スーパーコンピュータ上で CHFS クライアントに 1

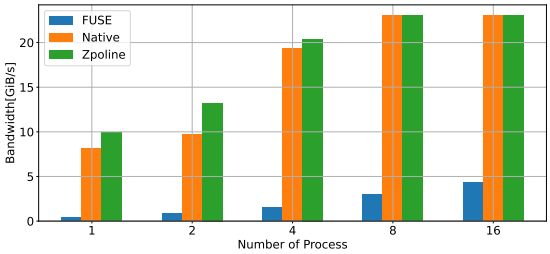

図 1: IOR file-per-process 読込み性能

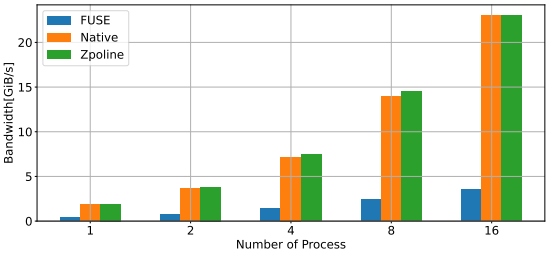

図 2: IOR file-per-process 書込み性能

ノード、CHFS サーバーに 10 ノード使用し、各ノードは 200Gbps の InfiniBand NDR200 で接続されている。計測には IOR ベンチマークを使用する。

図 1 で読込み性能を、図 2 で書込み性能を示す。zpoline を使用したシステムコールフックライブラリは、CHFS の API を直接呼び出した場合と同等の性能を示した。また FUSE を使用した場合と比較してより高い性能を示し、特に 16 プロセスにおいて FUSE の 5.3 倍高い読込み性能、6.4 倍高い書込み性能を示した。

5 結論

ユーザー空間における POSIX インターフェースのサポートにはいくつかの問題があったが、本研究により、zpoline を使用することでこれらの問題を解決できることが示された。提案したシステムコールフックライブラリは CHFS の API を直接呼び出した場合と同等の性能を示し、FUSE の 5.3 倍から 6.4 倍の性能結果を示した。

参考文献

[1] Osamu Tatebe, Kazuki Obata, Kohei Hiraga, and Hiroki Ohtsuji. Chfs: Parallel consistent hashing file system for node-local persistent memory. In *International Conference on High Performance Computing in Asia-Pacific Region*, pp. 115–124, 2022.

[2] Marc-André Vef, Nafiseh Moti, Tim Süß, Tommaso Tocci, Ramon Nou, Alberto Miranda, Toni Cortes, and André Brinkmann. Gekkofs - a temporary distributed file system for hpc applications. In *2018 IEEE International Conference on Cluster Computing (CLUSTER)*, pp. 319–324, 2018.

[3] Michael J. Brim, Adam T. Moody, Seung-Hwan Lim, Ross Miller, Swen Boehm, Cameron Stanavige, Kathryn M. Mohror, and Sarp Oral. Unifyfs: A user-level shared file system for unified access to distributed local storage. In *2023 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pp. 290–300, 2023.

[4] Kenichi Yasukata, Hajime Tazaki, Pierre-Louis Aublin, and Kenta Ishiguro. zpoline: a system call hook mechanism based on binary rewriting. In *2023 USENIX Annual Technical Conference (USENIX ATC 23)*, pp. 293–300, Boston, MA, July 2023. USENIX Association.

[5] LLNL. Gotcha. `https://github.com/LLNL/GOTCHA`.