

# Modulární komunikační reflektor s DV přenosem

Eva Hladká  
Fakulta informatiky,  
Masarykova univerzita v Brně,  
Botanická 68a, Brno 602 00,  
Czech Republic  
Email: eva@fi.muni.cz

Petr Holub  
Fakulta informatiky a  
Ústav výpočetní techniky,  
Masarykova univerzita v Brně,  
Botanická 68a, Brno 602 00,  
Czech Republic  
Email: hopet@ics.muni.cz

Miloš Liška  
Fakulta informatiky,  
Masarykova univerzita v Brně,  
Botanická 68a, Brno 602 00,  
Czech Republic  
Email: xliska@fi.muni.cz

**Abstrakt.** Přenosy videa s vysokým rozlišením generují datové toky v síti o řádu desítek Mb/s. V příspěvku je popsáno jak řešit přenosy videodat ve formátu Digital Video (DV) pro více než dva koncové body pomocí modulárního reflektoru, kde reflektor je softwarové zařízení pracující v uživatelském prostoru se základní vlastností multiplikovat a distribuovat data všem účastníkům komunikace. Popsaný reflektor byl implementován a článek uvádí výkonnostní parametry daného řešení a odhad škálovatelnosti.

**Klíčová slova:** skupinová komunikace, reflektor, video přenosy, DV formát, škálovatelnost

## I. ÚVOD

Současné vysokorychlostní sítě je možno využívat pro přenosy videa s vysokým rozlišením, což otevírá možnosti pro celou řadu nových aplikací využívajících vysílání jednosměrné (např. vysílání televize) a vícesměrové (např. aplikace pro vzdálenou spolupráci). Častým požadavkem je zaslání videa ne jednomu, ale celé skupině uživatelů. Ideálně škálujícím prostředkem pro takové přenosy jsou služby multicastu [1]. Tyto služby ovšem nepokrývají celý Internet a běžným jevem je buď naprosto chybějící podpora multicastu v lokální síti koncových uživatelů nebo špatná konfigurace aktivních prvků tak, že klient nemá možnost se účastnit všech probíhajících multicastových přenosů a Internet je tedy tvořen řadou podsítí, které mezi sebou multicast přenášejí buď omezeně nebo vůbec. Problémy spojené s provozem multicastu trvají více než desetiletí a proto i naděje na brzké řešení problémů šíření multicastového provozu je velmi malá. Tato skutečnost nás vedla k hledání alternativního řešení.

Možnou simulací nativní skupinové komunikace (multicastu) v případě, že požadavkem není neomezená škálovatelnost, je serializace paralelního schématu skupinové komunikace. Oproti multicastu s požadavkem replikace dat uvnitř sítě dle potřeby jsou v sériové simulaci stejné kopie dat z jednoho či více míst v síti rozepisovány postupně uživatelům. V tomto článku se budeme zabývat distribucí videa ve formátu Digital Video (DV) [2] skupině komunikujících účastníků s využitím simulace multicastu pomocí replikačního prvku nazvaného *reflektor* [3]. Architektura skupinové komunikace využívající

reflektory na jedné straně trpí do jisté míry eliminovatelnými problémy se škálovatelností, na straně druhé ale umožňuje řadu pokročilých služeb které jsou v nativním multicastovém prostředí prakticky nerealizovatelné a umožňuje také budovat *ad hoc* komunikační prostředí.

Článek je organizován následujícím způsobem: kapitola II pojednává o architektuře a základních vlastnostech reflektoru, kapitola III shrnuje pokročilé scénáře využití reflektoru, kapitola IV podává základní přehled přenosu videa ve formátu DV po IP sítích včetně popisu uživatelských nástrojů, kapitola V popisuje výkonnostní charakteristiky přenosu DV v prostředí založeném na reflektorech, kapitola VI je věnována možnému zlepšení škálovatelnosti vytvořením sítě reflektorů, kapitola VII shrnuje příbuzné technologie a kapitola VIII uzavírá problematiku přenosu DV v prostředí reflektorů.

## II. SIMULACE MULTICASTU POMOCÍ REFLEKTORU

Reflektor nazýváme síťový prvek, který replikuje a případně také zpracovává příchozí data typicky ve formě UDP datagramů pro přijímající klienty a tato data sekvenčně odesílá pouze s využitím unicastové síťové komunikace. V případě, že jsou data odesílána všem klientům, je počet kopií stejný jako počet přijímajících klientů. Náš návrh reflektoru vychází ze tří směrů vývoje v komunikačních sítích – překryvových sítí (overlay networks), uživatelem řízeného přístupu (user empowered approach) a programovatelných sítí (active networks). Reflektor je navržen jako uživatelem řízený modulární programovatelný směrovač s možností přisestavení modulů pro speciální zpracování přeposílaných datagramů. Reflektor pracuje pouze v uživatelském prostoru, což umožňuje běh bez nutnosti administrátorských oprávnění na hostitelském počítači.

Zpracování a replikace dat probíhá následujícím způsobem: data přijatá přijímajícími moduly jsou zařazena do fronty ke zpracování a vyhodnocena modulem modulem pro autentizaci, autorizaci a accounting. Jsou-li data tímto modulem autorizována k dalšímu zpracování, předají se modulu pro udržování sezení (session management), který si zaznamená aktivitu klienta v dané skupině, a dále jsou data zpracována dle

konfigurace reflektoru příslušnými moduly (procesory). Poté jsou data rozeslána klientům dané skupiny odesílacím modulem dle distribučního seznamu získaného z modulu pro udržování sezení. Z důvodu optimalizace výkonu je minimalizován počet vytvářených kopií dat a u jednoduchých scénářů reflektor pracuje v režimu zero-copy.

Modul pro udržování sezení je zodpovědný za přidávání nových klientů do komunikačních skupin (klient je zařazen do seznamu v okamžiku, kdy je od něj přijat první datagram) a za periodické odstraňování klientů, kteří se od reflektoru odpojili. Základní zabezpečení skupinové komunikace je řešeno pomocí modulu pro autentizaci, autorizaci a accounting (AAA), jenž umožňuje komunikující skupinu omezit na vybrané IP adresy a případně také nepřímo pomocí autentizace uživatele jménem a heslem. Není-li zkonfigurováno pomocí AAA modulu jinak, je kterémukoli klientovi umožněno se připojit ke skupině a stačí mu pouze začít zasílat datagramy na reflektor.

Reflektor obsahuje moduly poskytující administrativní rozhraní přes zabezpečené protokoly jako je HTTP s podporou SSL/TLS nebo SOAP s podporou GSI [4]. Komunikace mezi klientem a administrativním rozhraním probíhá pomocí jazyka RAP [5]. Toto rozhraní umožňuje klientům po úspěšné autentizaci a autorizaci řídit chování reflektoru za běhu: je možné měnit autorizační informace pro přeposílání datagramů, nahrávat, zastavovat a restartovat jednotlivé moduly reflektoru či modifikovat jejich chování.

Virtuální multicast implementovaný pomocí centrálního reflektoru přináší problémy s omezenou škálovatelností. Tyto problémy lze částečně řešit pomocí sítí reflektorů propojených mezi sebou tunely, které mohou být vytvářeny buď staticky nebo dynamicky (např. pomocí směrovacích algoritmů typu distance vector nebo na bázi některých efektivnějších směrovacích algoritmů peer-to-peer sítí jako Pastry [6] apod.). Příklad takového řešení je uveden v kapitole VI. Sítě reflektorů mohou být také využity pro vytvoření prostředí, které je rezistentnější vůči výpadkům než běžná síť [7].

### III. POKROČILÉ VLASTNOSTI SIMULACE MULTICASTU S VYUŽITÍM REFLEKTORŮ

Díky replikaci dat pro každého klienta zvláště je možné implementovat zpracování odlišné pro různé klienty a modularita reflektoru umožňuje moduly s pokročilejšími funkcemi přidávat a konfigurovat za běhu reflektoru. V následujících odstavcích jsou uvedeny příklady takového zpracování.

a) *Změna kódování multimediálních dat:* Moduly zpracovávající data mohou transformovat data mezi různými formáty (např. převádět video ve formátu DV do formátu H.261). Reflektor tedy může sloužit jako brána, která umožňuje připojení klientům s omezenou podporou kompresních formátů, s nedostatečnou výpočetní kapacitou či pro klienty, kteří jsou připojeni pomocí pomalých síťových linek, a přitom nezabraňuje zbytku sku-

piny komunikovat pomocí formátů s vyšší kvalitou a náročností.

b) *Skládání několika obrazů:* Skládání obrazu z několika souběžných zdrojů videa je užitečné například v případě kolaborativního prostředí s větším množstvím účastníků, kdy na koncových stanicích již není dostatečná výpočetní a zobrazovací kapacita pro zobrazování mnoha proudů videa od všech klientů současně. Je také možno propojit tunelem reflektor s funkcí skládání obrazů a reflektor bez této funkce a připojit klienty dle kapacity jejich připojení. To je jedna z variant skupinové komunikace klasickým multicastem neřešitelná.

c) *Synchronizace:* V případě paralelních proudů dat enkapsulovaných do protokolu RTP [8] je možno provádět mezi těmito proudy synchronizaci [9]. RTP pakety obsahují relativní časové značky které mohou být převáděny na absolutní čas na odesílající stanici pomocí časových značek v relativním i absolutním čase uvedených v komplementárních RTCP paketech. Provádíme-li synchronizaci mezi zdroji z různých odesílajících počítačů, je třeba synchronizovat jejich čas např. pomocí protokolu NTP.

Využitím pokročilých vlastností virtuálního multicastu na bázi reflektorů je možno také vytvořit silně zabezpečené kolaborativní prostředí, kdy každý klient udržuje s reflektorem spolehlivý zabezpečený kanál (TCP kanál šifrovaný pomocí SSL) pomocí něhož jsou vyměňovány šifrovací klíče mezi klientem a reflektorem. UDP datagramy jsou pak šifrované odesílány z klienta na reflektor, kde jsou zpracovány a opět zašifrované odeslány na přijímající klienty. Takto upravený reflektor spolu s modifikovanými klientskými nástroji MBone Tools byl studován v [10].

Reflektor je možné využít i v síťovém prostředí omezeném firewallem a překladem adres (NAT), kde lze využít tunelování UDP datagramů pomocí TCP spojení přes některý z povolených portů [3], [11].

## IV. PŘENOS VIDEO VE FORMÁTU DV S VYUŽITÍM REFLEKTORU

Digital Video (DV) [2], [12] je v současnosti jeden z běžných formátů pro kompresi a ukládání digitálního videa. Jeho výhodou je vysoká kvalita v plném rozlišení (např. pro normu PAL se jedná o rozlišení  $720 \times 576$  se snímkovou frekvencí 25 snímků/s), velmi malá degradace obrazu vlivem komprese, a to i při několikanásobné kompresi v důsledku vícenásobného zpracování, a také relativně snadná dostupnost zařízení, které s DV formátem dokáží přímo pracovat pomocí rozhraní IEEE-1394 (kamery, převodníky, rekordéry atd.). DV komprese používá vzorkování 4:1:1 pro normu NTSC a 4:2:0 pro normu PAL a využívá pouze intra-frame kompresi velmi podobnou kompresi MPEG s pevně stanoveným počtem bitů za sekundu. Zvuk se kóduje společně s příslušným rámcem videa. Vzorkovací frekvence zvukové stopy je 32 kHz, 44,1 kHz nebo 48 kHz a kvantování využívá 12, 16 nebo 20 bitů. Kvalita zvuku tedy může být vyšší než kvalita záznamu na CD a limitujícím faktorem obvykle

TABULKA I  
Konfigurace testovacího prostředí

	test4	brand	gerard
značka/model	–	DELL PowerEdge 1600SC	DELL PowerEdge 1600SC
procesor	2× Intel Xeon 2.80 GHz	2× Intel Xeon 2.80 GHz	2× Intel Xeon 2.80 GHz
paměť	1024 MB	1024 MB	1024 MB
síťová karta	Intel 82545EM 64 bit/66 MHz	Broadcom BCM5701 64 bit/100 MHz	Intel PRO/1000 32 bit/66 MHz
operační systém	Linux 2.4.23	FreeBSD 5.2-RELEASE	FreeBSD 5.2-RELEASE

bývá kvalita akvizčních zařízení. Díky skutečnosti, že DV nevyužívá inter-frame kompresi, je možné pomocí formátu DV také realizovat přenosy s nízkou latencí – při využití vhodných koncových zařízení je možné latenci snížit na přibližně 100 ms [13] (při využití nekomprimovaného analogového videa se latence pohybuje kolem 70 ms).

Přenos DV v IP sítích je specifikován v RFC 3189 a RFC 3190 a probíhá nad UDP datagramy prostřednictvím protokolu RTP. Čistě softwarová implementace přenosu DV v IP sítích byla realizována v rámci projektu DVTS<sup>1</sup> [14], [15]. V rámci tohoto projektu byly vytvořeny nástroje pro odesílání a příjem DV pro operační systémy Linux, Free/Net/OpenBSD a Windows a také prototypové a nepříliš stabilní aplikace pro zobrazování DV v prostředí MS Windows a X Windows (xdvshow).

Vzhledem k problémům se stabilitou zobrazovacích nástrojů z projektu DVTS jsme se rozhodli vytvořit na základě programu xdvshow vlastní softwarovou implementaci pro operační systémy Linux a FreeBSD. Ta umožňuje zobrazování videa jak ve formátu PAL tak i NTSC pomocí knihovny libdv [16] dostupné jako open-source produkt<sup>2</sup>. Naše implementace využívá robustní vícevláknovou architekturu, která separuje příjem dat ze sítě od renderování a zobrazování samotného obrazu. Zobrazování je možné buď čistě přes rozhraní X11, nebo s využitím SDL knihovny [17]. V případě použití SDL je možné přehrávat video také v celoobrazovkovém režimu, a to buď škálovaně interpolací na vyšší rozlišení, nebo neškálovaně s vyplněním zbytku obrazovky černým rámem v případě nedostatečného výpočetního výkonu použitého hardware.

Tato nová implementace také podporuje spolupráci s výše popsaným reflektorem. Protože zobrazovací klient je pouze pasivní a odesílání DV je prováděno jiným nástrojem, jsou reflektoru zasílány v pravidelných časových intervalech RTCP pakety na port, který je o jedničku vyšší než port s DV streamem zabaleným v RTP tak, aby mohl reflektor provádět údržbu sezení (session management) a případně mohl od pasivních klientů sbírat statistické údaje o kvalitě příjmu.

<sup>1</sup>Další rozvoj standardu přenosu DV v IP sítích je nyní zajišťován pomocí organizace DVTS Consortium, v níž nyní participuje i Masarykova univerzita v Brně. Projekt DVTS vyvíjející samotný software však ustrnul ve vývoji z důvodu odchodu hlavního vývojáře.

<sup>2</sup>Původní implementace xdvshow z projektu DVTS umožňuje pouze zobrazování videa v NTSC formátu.

TABULKA II

Výkonnost reflektoru pro zvyšující se počet DV klientů.

# klientů	Výstupní vytížení reflektoru [Mb/s]	
	teoretické	experimentální
1 (+1)	30	29.913 ± 0.027
2 (+1)	120	118.21 ± 0.12
3 (+1)	270	270.6 ± 2.1
4 (+1)	480	480.2 ± 2.8
5 (+1)	750	742.0 ± 4.2
6 (+1)	1080	908.2 ± 2.4
7 (+1)	1470	916.0 ± 1.2

## V. EXPERIMENTÁLNÍ VÝSLEDKY

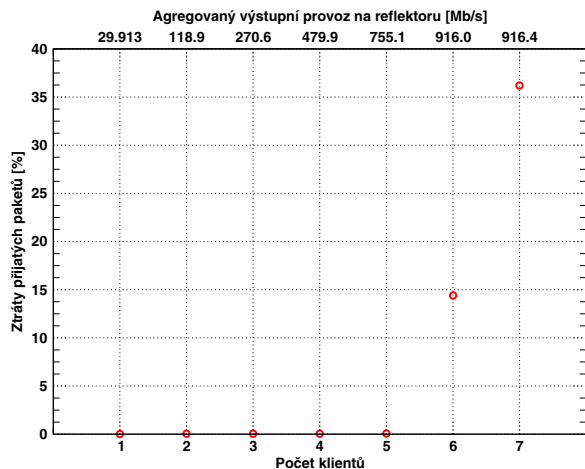
Výše uvedený reflektor byl implementován a pro ověření použitelnosti jsme provedli základní měření výkonnosti. Testovací prostředí se sestávalo ze tří výkonných počítačů, které fungovaly jako generátor provozu (gerard), reflektor (test4) a přijímač (brand), spojené pomocí gigabitového prepínače HP ProCurve 6108. Konfigurace počítačů jsou uvedeny v tab. I

Výkonnost reflektoru v kombinaci s DV klienty byla měřena jako závislost výkonu reflektoru na počtu klientů odesílajících a přijímajících DV data s rychlostí 30 Mb/s. V průběhu experimentu se zvyšoval počet aktivních klientů a v průběhu celého experimentu zůstával jeden pasivní klient (pouze přijímající klient), který současně sloužil jako měřicí sonda. Výsledky shrnuté v tab. II a na obr. 1 a 2 ukazují, že systém je použitelný pro spolupráci až pěti klientů pracujících s vysoce kvalitním videem ve formátu DV.

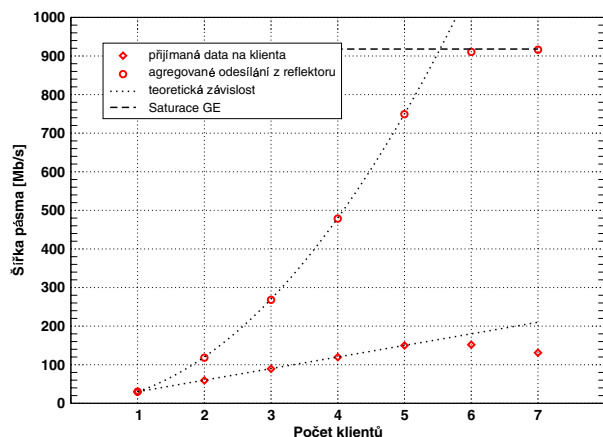
Z výsledků je zřejmé, že na výkonném PC je reflektor plně schopen saturovat gigabitové ethernetové připojení v rámci možností daných architekturou PC a daným operačním systémem. Z výsledků je také zjevná omezená škálovatelnost prostředí založeného na reflektorech, kterou je možné řešit pomocí vytvoření sítě reflektorů (kap. VI).

Dále jsme otestovali výkonnost reflektoru v čistě přesílalajícím (forward) režimu, který se ukazuje jako náročnější pro zpracování než běžný replikační mód reflektoru. To odpovídá skutečnosti, že je více dat přenášeno přes PCI sběrnici než v případě replikačního režimu. Výsledky shrnuté graficky na obr. 3 ukazují, že prakticky

Obr. 1. Závislost počtu DV klientů na ztrátách na reflektoru



Obr. 2. Charakteristiky reflektoru v závislosti na počtech klientů



bez výpadků je schopen reflektor v tomto režimu pracovat do průtoku 450 Mb/s.

## VI. ŠKÁLOVATELNOST

Jak je zmíněno v kap. II a V, škálovatelnost a výkonnost komunikačního prostředí založeného na reflektorech lze zlepšit využitím více reflektorů propojených pomocí tunelů. Nejjednodušší model, který současně může sloužit jako odhad nejhorší efektivity, je vytvoření úplného grafu, tak že každý reflektor komunikuje přímo s každým, jak je znázorněno na obr. 4.

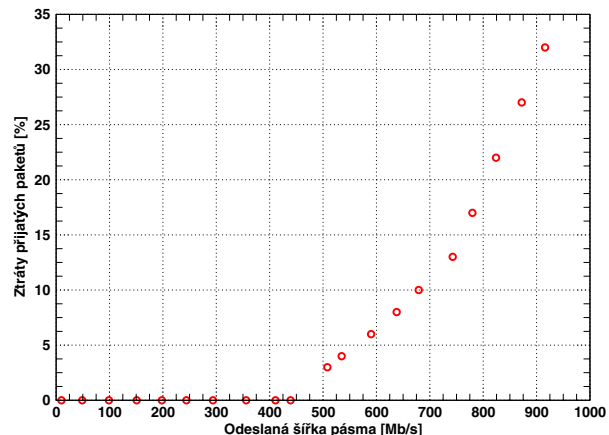
Předpokládáme síť reflektorů, v níž je ke každému reflektoru připojeno  $n_r$  nebo  $n_r - 1$  klientů (jedná se o co nejrovnoměrnější rozdělení klientů mezi reflektory). Lze odvodit, že počet vstupních toků každého reflektoru je

$$in = n, \quad (1)$$

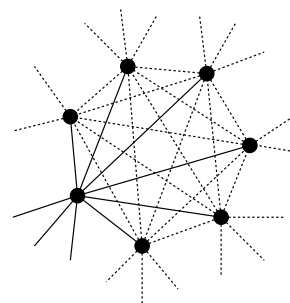
počet výstupních toků pro reflektor s  $n_r$  klienty je

$$out_{n_r} = n_r(m + n - 2), \quad (2)$$

Obr. 3. Čistě přeposílací propustnost reflektoru



Obr. 4. Full-mesh model tunelování mezi reflektory.



kde  $m$  je počet reflektorů a  $n$  je celkový počet připojených klientů. Poměr počtu výstupních toků mezi reflektory s  $n_r$  a  $n_r - 1$  klienty je

$$\frac{out_{n_r-1}}{out_{n_r}} = \frac{n_r - 1}{n_r}. \quad (3)$$

Vezmeme-li v úvahu, že  $n_r = \lceil \frac{n}{m} \rceil$  a počet výstupních proudů na samostatně použitém reflektoru je  $out_s = n(n - 1)$ , lze odvodit, že poměr mezi počtem výstupních toků v síti reflektorů  $out_{n_r}$  a samostatně stojícím reflektorem  $out_s$  je přibližně

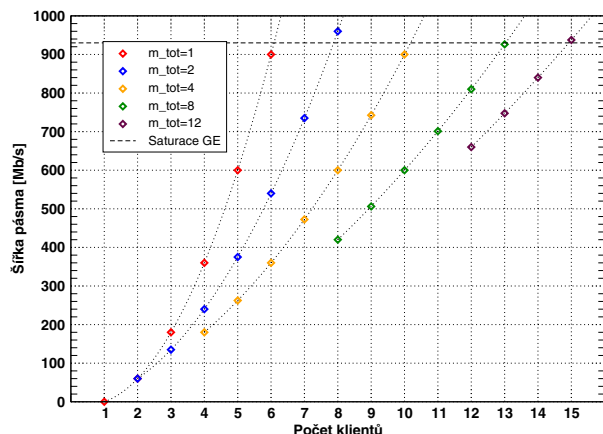
$$\frac{out_{n_r}}{out_s} \approx \frac{m + n - 2}{m(n - 1)} \quad (4)$$

Z grafu ilustrujícího maximální výstupní tok na jeden reflektor pro síť s různým počtem reflektorů (obr. 5) je zřejmé, že z původních maximálně šesti DV klientů při použití jednoho reflektoru a gigabitového připojení se kapacita prostředí zvýšila až na 15 účastníků při použití síti 12 reflektorů s gigabitovým připojením.

## VII. PŘÍBUZNÉ PRÁCE

Řešení problémů skupinové komunikace (multicast) pomocí různých typů přemostění a reflektorů má dlouhou historii. Snad nejlépe je to vidět na videokonferenčním systému VRVS [18], který střídavě využíval sítě zrcadel a multicastu.

Obr. 5. Závislost výstupního datového toku na reflektorech v závislosti na počtu DV klientů pro konfigurace sítí s počtem reflektorů  $m = 1, 2, 4, 8, 12$ .



Původní návrh reflektoru popisovaného v příspěvku navazoval na práci J. Higfielda [19]. Jednalo se velmi jednoduchý program, jehož jedinou funkcí bylo poslouchat na předem určeném portu UDP datagramy a přeposílat je všem aktivním účastníkům komunikace (těm, kteří během určeného intervalu nějaká data zaslali).

Systém VRVS využívá pro skupinovou komunikaci pevně danou síť zrcadel. Komunikující jsou připojeni ke geograficky nejbližšímu zrcadlu bez možnosti volby a případné kontroly. Zrcadla jsou poskytována jako služba, bez možnosti zásahu do kódu či administrace. Unikátním rysem je možnost propojení H.323 a MBone videokonferencí.

AccessGridové videokonference [20] využívají pro skupinovou komunikaci nativní multicast, ale pro místa bez multicastové konektivity byly vytvořeny dva typy reflektorů: QuickBridge [21] a Multi-Session Bridge [22]. QuickBridge pracuje jako spoj do multicastové sítě. Je instalován uvnitř MBone a klient vně se na něj připojuje unicastem. Je to jednoduchý reflektor využívající autentizaci pomocí IP adresy a pracující s databází AG bodů a odpovídajících multicastových adres a portů. Multi-Session Bridge je obdobný systém provozovaný na centrálních AccessGrid serverech v Argonne National Laboratory.

### VIII. ZÁVĚR

Kombinace reflektoru s přenosem DV může být využita pro videokonferenční aktivity vyžadující vysokou kvalitu obrazu. Příkladem mohou být telemedicínské přenosy operací či obrazu z kamery připojené k mikroskopu, kde maximální kvalita obrazu umožňuje lékařům stanovovat přesněji a spolehlivěji diagnózu. Další aplikací v oblasti vzdělávání může být vysílání přednášek mezi několika geograficky vzdálenými poslucháři v reálném čase s využitím IP sítí.

Zajímavou aplikací může být přenos stereoskopického videa ve formátu DV. Zde je možno využít synchronizační schopnosti reflektoru a posílat videodata samo-

statně pro každé oko, což umožňuje následné náročnější zpracování obrazu, než je možné v případě jednoho kombinovaného obrazu.

### PODĚKOVÁNÍ

Tato práce byla řešena za podpory Výzkumného záměru Optická síť národního výzkumu a její nové aplikace MSM 6383917201. Autoři by také rádi poděkovali Jiřímu Denemarkovi a Tomáši Rebokovi z FI MU za pomoc při implementaci reflektoru a měřicích nástrojů.

### LITERATURA A ODKAZY

- [1] R. Wittmann, M. Zitterbart. *Multicast communication: protocols, programming, and applications*. Morgan Kaufmann Publishers, 1999.
- [2] International Electrotechnical Commission. *IEC 61834: Recording – Helical-scan digital video cassette recording system using 6.35 mm magnetic tape for consumer use (525-60, 625-50, 1125-60 and 1250-50 systems)*, 1998, 1999, 2001. Parts 1–10, <http://www.iec.ch>.
- [3] Ěva Hladká, Petr Holub, Jiří Denemark. "User empowered virtual multicast for multimedia communication," v *Proceedings of ICN 2004*, 2004.
- [4] "GSI SOAP," <http://www.doesciencegrid.org/Grid/projects/soap/>.
- [5] Jiří Denemark, Petr Holub, Ěva Hladká. "RAP - Reflector Administration Protocol," technická zpráva 9/2003, CESNET, 2003.
- [6] A. Rowstron, P. Druschel. "Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems," v *IFIP/ACM International Conference on Distributed Systems Platforms (Middleware)*, Heidelberg, Germany, 2001, pp. 329–350.
- [7] David Andersen, Hari Balakrishnan, Frans Kaashoek, Robert Morris. "Resilient overlay networks," v *18th ACM Symp. on Operating Systems Principles (SOSP)*, Banff, Canada, Oct. 2001.
- [8] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson. "RTP: A Transport Protocol for Real-Time Applications," RFC 3550, July 2003.
- [9] Tomáš Rebok, Petr Holub. "Synchronizing RTP Packet Reflector," technická zpráva 7/2003, CESNET, 2003.
- [10] Tomáš Bouček. "Kryptografické zabezpečení videokonferencí," diplomová práce, Vojenská akademie Brno, 2002.
- [11] Zdeněk Salvat. "Enhanced UDP packet reflector for unfriendly environments," technická zpráva 16/2001, CESNET, 2001.
- [12] *Specifications of Consumer-Use Digital VCRs using 6.3mm magnetic tape*. Video Equipment Division, AVC Company, Matsushita Electric Industrial Co., Ltd. 1–15, Matsuo-Cho, Kadoma-Shi, Osaka, 571-8504, JAPAN, Dec. 1994. unofficially called Blue Book.
- [13] "McGill Ultra-Videoconferencing System: FAQ," <http://ultravideo.mcgill.edu/faq/>.
- [14] Akimichi Ogawa, Katsushi Kobayashi, Kazunori Sugiura, Osamu Nakamura, Jun Murai. "Design and implementation of DV based video over RTP," v *Packet Video Workshop 2000*, 2000. <http://www.sfc.wide.ad.jp/DVTS/pv2000/index.html>.
- [15] Akimichi Ogawa. "DVTS: Digital video transport system," 1999-2002. <http://www.sfc.wide.ad.jp/DVTS/>.
- [16] Charles Krasic, Erik Walthinsen. "Quasar dv codec: libdv," <http://libdv.sourceforge.net/>.
- [17] "SDL: Simple Directmedia Layer," <http://www.libsdl.org/>.
- [18] P. Galvez, G. Denis, H. Newman. "Networking, Videoconferencing and Collaborative Environments," v *Proceedings of CHEP'98*, Chicago, September 1998.
- [19] Julian Highfield. "UDP packet reflector hacks – RTP unicast mirror rum,"
- [20] L. Childers, T. Disz, R. Olson, M. E. Papka, R. Stevens, T. Udeshi. "Access grid: Immersive group-to-group collaborative visualization," v *Proceedings of Immersive Projection Technology*, Ames, Iowa, 2000.
- [21] M. Daw. "How to install and use a quick multicast bridge (QuickBridge),"
- [22] G. A. Roediger. "The Multi-Session Bridge," <http://www.hep.net/chep98/PDF/230.pdf>.