

Jacek PIASECKI

Politechnika Śląska, Instytut Informatyki

DOSTĘP DO ZASOBÓW LOKALNYCH I SIECIOWYCH SYSTEMÓW KOMPUTEROWYCH ZE STANOWISK ODLEGŁYCH

Streszczenie. W publikacji przedstawiono problemy związane z dostępem do zasobów lokalnych i sieciowych systemów komputerowych ze stanowisk odległych, a w szczególności problem związany z szybkością przesyłu zawartości pamięci obrazu (o różnym rozmiarze wynikającym z trybu karty graficznej), oraz przygotowania danych do transmisji w powiązaniu z przepustowością zastosowanego łącza transmisyjnego.

THE REMOTE ACCESS TO RESOURCES OF LOCAL AND NETWORK COMPUTER SYSTEMS

Summary. In the paper some problems connected with the remote access to resources of the local and network computer systems were presented. We focussed on the display memory transfer rate and on preparing data to transmission, both with respect to the throughput of the transmission medium.

1. Wprowadzenie

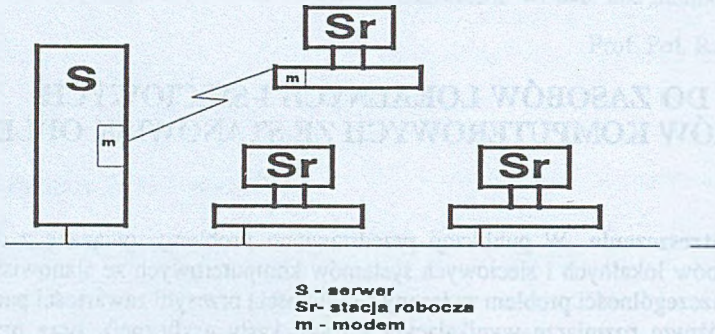
W pewnym obszarze zastosowań przemysłowych sieciowych systemów sterowania istnieje potrzeba kontroli aplikacji na odległość celem monitorowania prawidłowości działania zarówno procesu, jak i działania systemu sterującego. Narzędzia umożliwiające taką pracę powinny zapewniać: pełne monitorowanie, sterowanie oraz modyfikację oprogramowania.

Istnieją dwa rodzaje (klasy) rozwiązań problemu dostępu do sieci ze stanowisk odległych:

- zdalny dostęp (remote access);
- zdalna kontrola (remote control).

Rozwiązania te różnią się między sobą w istotny sposób i nie zawsze nadają się do tych samych zastosowań [10].

Idea zdalnego dostępu (rys. 1.) oparta jest na zasadzie udostępnienia stacji odległej pełnych zasobów sieci i traktowania tej stacji jak każdej innej stacji roboczej, wykorzystując do tego celu łącze telefoniczne. Przykładem może być usługa RAS (Remote Access Services) dostępna w środowisku Windows NT [9].



Rys. 1. Zdalny dostęp do sieci

Fig. 1. Remote access

Na serwerze, z systemem operacyjnym Windows NT, musi być zainstalowana i skonfigurowana usługa zdalnego dostępu, natomiast na stacji odległej musi być zainstalowane oprogramowanie w wersji stacji roboczej. Serwer NT może obsługiwać do 256 kanałów modemowych. W przypadku obsługi przez serwer większej liczby kanałów modemowych zauważalny staje się spadek wydajności systemu, jeśli serwerem jest komputer klasy PC. Osobnym problemem są fizyczne możliwości wbudowania większej liczby kart modemowych do tego typu sprzętu. Oczywiście, system operacyjny Windows NT może być zainstalowany w maszynach wieloprocesorowych, jednakże zastosowanie jako serwera komputera wieloprocesorowego jest rozwiązaniem znacznie kosztowniejszym [9] [3].

Usługa zdalnego dostępu (RAS) umożliwia między innymi:

- dostęp do sieci za pomocą trzech podstawowych protokołów transmisyjnych: TCP/IP, IPX, oraz NetBIOS;
- realizację usługi callback (oddzwanianie - serwer nawiązuje połączenie ze stacją roboczą);
- przesyłanie hasła i nazwy użytkownika w postaci zakodowanej przy nawiązywaniu połączenia.

Istnieją i inne systemy zapewniające zdalny dostęp, jednakże ogólne zasady pracy pozostają takie same. Administrowanie tego typu dostępem nie jest skomplikowane i

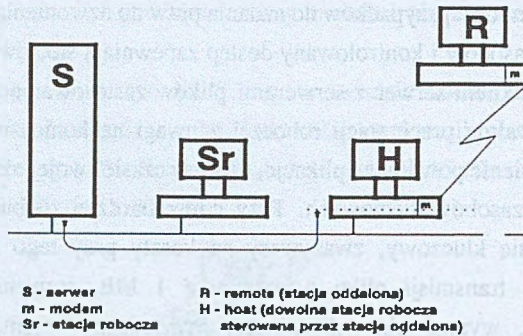
ogranicza się w większości przypadków do nadania praw do dzwonienia przez użytkowników, natomiast ochronę zasobów i kontrolowany dostęp zapewniają sieciowe systemy operacyjne.

W sieciach typu klient-serwer z serwerami plików zastosowanie łącza szeregowego w istotny sposób spowalnia pracę stacji roboczej z uwagi na konieczność ściągania plików. Dodatkowe spowolnienie powodują aplikacje, które w czasie swojej aktywności muszą często odwoływać się do zasobów sieciowych. Przy coraz bardziej rozbudowanych aplikacjach problem ten staje się kluczowy, zważywszy na koszty przy tego rodzaju pracy (opłaty telefoniczne). Czas transmisji pliku o rozmiarze 1 MB, transmitowanego ze średnią prędkością 4 kB/s, wynosi ok. 4,5 minuty. Przytoczony orientacyjny czas może w rzeczywistości być krótszy z uwagi na działanie protokółów kompresji danych (w powyższym przykładzie nie uwzględniłem narzutów wynikających z zastosowanego protokołu oraz ewentualnych retransmisji). Dodatkowym utrudnieniem podczas pracy jest przerwanie połączenia w kanale, co prowadzi do automatycznego "wylogowania" stacji roboczej (cecha niektórych typów protokółów, lub jest pożądaną z punktu widzenia bezpieczeństwa zasobów użytkownika, czy sieci). Wznowienie pracy jest możliwe po ponownym zainicjowaniu połączenia i próbie uruchomienia aplikacji. Jeżeli w poprzedniej sesji (przerwanej) nie zostały zachowane np. dokument, stan aplikacji itd., to praca w czasie tej sesji była czasem straconym.

Osobnym zagadnieniem jest zastosowanie zdalnego dostępu do sterowania i monitorowania odległych procesów przemysłowych lub pomiarowych z uwagi na bezpieczeństwo przebiegu tych procesów (zmiana istotnych parametrów sterowania, reakcja na stany awaryjne itp.). W istocie zastosowanie zdalnego dostępu spowodowałoby zdublowanie systemu monitoringu i sterowania. Należałoby zapewnić prawidłową pracę takich instalacji, uwzględniającą różne czasy obsługi procesów, wynikającą z szybkości transmisji (zakładając, że jeden z systemów sterowania pracuje w sieci, a drugi poprzez np. łącze telefoniczne), rozwiązać problem sterowalności procesu w przypadku błędów lub zrywania połączenia itp.

Przy zdalnej kontroli (rys. 2.) odległa stacja nie pracuje bezpośrednio w sieci jako stacja robocza, lecz korzysta z usług gospodarza (hosta) i za jego pośrednictwem ma dostęp do zasobów sieciowych. Odległy komputer kontroluje pracę gospodarza, czyli do hosta przesyła kody klawiatury, natomiast w drugą stronę przesyłana jest zawartość pamięci obrazu. Wybrane procesy w trakcie połączenia uruchamiane są na komputerze pełniącym rolę gospodarza i jeśli host pracuje w sieci, to on jest stacją roboczą. Przykładem tego rozwiązania może być Norton pcAnywhere firmy Symantec [2].

W przeciwieństwie do poprzedniej metody, metody zdalnego dostępu, przy zdalnej kontroli nie istnieje konieczność pracy z wymaganym w danej sieci protokołem na drodze odległy komputer - sieć (odległy komputer nie jest stacją roboczą pracującą w sieci, a tylko



Rys. 2. Przykład systemu komputerowego z realizacją zdalnej kontroli
 Fig. 2. The example of a computer system with remote control

ekranem i klawiaturą hosta). Podczas pracy czasy oczekiwania są związane tylko z czasem transmisji ekranu, znaków z klawiatury i są znaczące w stosunku do czasu traconego na transmisję host - serwer, w przypadku kiedy host jest stacją roboczą. Większość narzędzi oferujących ten typ usługi nie wprowadza ograniczeń związanych ze sterowaniem hosta oraz korzystania z usług sieciowych. Pewnym problemem może być ochrona zasobów lokalnych kontrolowanego komputera oraz kontrola dostępu do hosta. O ile od strony sieci powyższy problem rozwiązuje sieciowy system operacyjny, to zasoby hosta musi chronić oprogramowanie zdalnej kontroli. W istocie większość narzędzi do zdalnej kontroli zapewnia:

- możliwość ustawienia atrybutów dostępu do zasobów zarówno lokalnych, jak i sieciowych dla każdego użytkownika z osobna;
- blokadę klawiatury hosta i wygaszanie ekranu hosta (zapewnienie poufności pracy);
- restart hosta po zakończeniu pracy lub zerwaniu połączenia;
- zabezpieczenie dostępu hasłem;
- szyfrowanie hasła;
- limitowanie czasu pracy użytkownika.

Ponadto istnieje możliwość realizacji wydruków na stanowisku odległym oraz korzystania z innych urządzeń zewnętrznych, jednak związane są z tym dodatkowe transmisje oraz zwiększona zajętość pamięci operacyjnej zarówno stacji host, jak i stacji odległej.

W porównaniu ze zdalnym dostępem, zdalna kontrola jest bardziej przydatna do monitorowania i sterowania odległych procesów, gdyż: nie wymaga rozbudowywania aplikacji sterujących (w większości przypadków wystarcza zainstalowanie odpowiedniego, prostego oprogramowania komunikacyjnego na stacji sterująco-monitorującej, oraz na stacji odległej). Metoda zdalnej kontroli nie wprowadza żadnych ograniczeń co do typu zastosowanej sieci, jak i rodzaju protokołu transmisyjnego. Natomiast metoda ta zwiększa obciążenie procesora kontrolowanego komputera, co jest związane z obsługą urządzenia

komunikacyjnego oraz przetwarzaniem zawartości pamięci obrazu. Problemem jest i będzie transmisja ekranu, jeżeli tego typu dostęp spełnić ma warunek elastyczności i uniwersalności wobec kontrolowanych aplikacji. Z tym wiąże się wiele problemów będących wynikiem rozmiaru informacji koniecznej do przesłania łączem transmisyjnym. Dodatkowo, istotną sprawą jest problem bezpieczeństwa, który pozornie kłóci się z uniwersalnością i elastycznością, podstawowym postulatem stawianym dla zdalnej kontroli jako narzędzia umożliwiającego sterowanie i monitoring na odległość dowolnego procesu przemysłowego, kontrolno-pomiarowego i innych.

2. Analiza istniejących problemów związanych z zastosowaniem zdalnej kontroli

Pierwszym problemem jest wejściowa wielkość danych konieczna do przesłania do stacji odległej, celem uzyskania obrazu, co w tym przypadku jest elementem koniecznym, aby móc wykonywać funkcje kontrolno-sterujące.

W celu uzyskania obrazu można:

- przysyłać odwołania do karty graficznej również kanałem transmisyjnym, jednakże z uwagi na brak kompatybilności sprzętu nie można mieć gwarancji, że otrzymamy dokładnie ten sam obraz z drugiej strony, jakiegokolwiek przekłamanie w trakcie transmisji może całkowicie uniemożliwić prawidłowe powielenie ekranu na stacji odległej;

- przesłać zawartość pamięci ekranu, tutaj jednak zaczyna odgrywać rolę wielkość informacji koniecznej do transmisji, przy czym wielkość ta jest zależna od rodzaju ekranu wynikającego z trybu pracy danej karty graficznej. Rozmiar danych reprezentujących ekran dla danego trybu karty graficznej można obliczyć ze wzoru:

$$D_v = \frac{R_x \cdot R_y \cdot d}{8} \quad (1)$$

gdzie: D_v - minimalny rozmiar pamięci wideo w bajtach;

R_x - rozdzielczość pozioma (liczba kolumn);

R_y - rozdzielczość pionowa (liczba wierszy);

d - ilość bitów kodujących daną liczbę kolorów.

Przykładowe rozmiary minimalne pamięci wideo dla wybranych trybów kart graficznych przedstawia tabela 1 [4] [5].

Na podstawie danych z powyższej tabeli należy wyciągnąć następujące wnioski:

- najmniejsze rozmiary pamięci obrazu zapewnia tryb znakowy (4 kB), jednakże jest nieprzydatny do celów graficznej wizualizacji procesów;
 - dla trybów graficznych pamięć obrazu zajmuje min 154 kB i szybko rośnie wraz ze wzrostem rozdzielczości, jak i wzrostem liczby kolorów.
- Kolejnym problemem jest wybór łącza transmisyjnego. Bardzo duże szybkości transmisji zapewniają lokalne sieci komputerowe oparte np.: na sieci typu Ethernet. Jednak metoda dostępu ze stanowiska odległego nie może być ograniczona tego typu rozwiązaniami z uwagi

Tabela 1

Przykładowe zapotrzebowanie na pamięć obrazu

Tryb	Rozdzielczość	Kolory	Pamięć obrazu
znakowy	80 x 25 znaków	16	4 kB
graficzny	640 x 480	16	153.6 kB
graficzny	1280 x 1024	65536	2.63 MB

na ich lokalny charakter [1] [11].

Pod względem liczby zastosowań, w omawianych rozwiązaniach, zdecydowany prym wiodzie sieć Internet (dostęp z poziomu sieci lokalnej lub łącza telefonicznego), oraz samo łącze telefoniczne (analogowe lub cyfrowe). Problemem jednak jest szybkość transmisji przy tego typu rozwiązaniach. W przypadku Internetu, pomimo stosowania coraz to szybszych łączy (szybsze łącze poprawia sytuację na stosunkowo krótki okres czasu), nie można zapewnić szybkiego transferu danych z uwagi na ciągły wzrost ruchu w sieci, średnio 10-krotnie w stosunku rocznym. Dodatkowo ruch w sieci wykazuje olbrzymią dynamikę zmienności w czasie, co w konsekwencji prowadzi przede wszystkim do również dynamicznie zmieniającej się chwilowej szybkości transmisji.

Połączenie dwóch komputerów bezpośrednio poprzez łącze telefoniczne ma swoje zalety i wady. Analogowe łącze telefoniczne ze względu na swoją specyfikę zapewnia stosunkowo małą szybkość transmisji 33.6 Kb/s (57.6 Kb/s). W dodatku łącze to charakteryzuje się dużym poziomem zakłóceń, konsekwencją czego są częste retransmisje prowadzące do obniżenia prędkości transmisji. W przeciwieństwie do wcześniej omawianych rodzajów sieci zapewnia jednak całe dostępne pasmo na cały czas trwania transferu. Łącze telekomunikacyjne cyfrowe (ISDN) zapewnia teoretycznie znacznie większe prędkości transmisji (2 Mb/s), jednak w bazowym pakiecie (2B+D) szybkość transmisji ograniczona

jest do 144 kb/s [12]. Dodatkowo nie można o tym łączu powiedzieć, że jest to łącze już powszechnie dostępne. Parametry wybranych łączy przedstawia tabela 2.

Szybkość transmisji na wybranych łączach

Tabela 2

Łącze	Szybkość transmisji	Uwagi
Lokalne sieci komputerowe	osiągane są bardzo duże szybkości transmisji powyżej 1 Mb/s	dostęp ograniczony zasięgiem sieci lokalnej
Internet	od kilkunastu Kb/s do kilkuset b/s	zmienna w czasie szybkość transmisji
Telefoniczne: analogowe	na poziomie kilkudziesięciu kb/s	powszechnie dostępne, łącza o niskiej jakości
ISDN	podstawowy pakiet 2B+D 144 kb/s	mało dostępne lub niedostępne na obszarach nisko zurbanizowanych
komórkowe	bardzo mała (9600 b/s)	ograniczone zasięgiem sieci komórkowej

Mając na względzie przede wszystkim ilość transmitowanych danych, należałoby wykorzystywać do zdalnej kontroli łącze zapewniające jak największą szybkość transmisji. Jednakże tylko sieć Internet i łącze telefoniczne zapewniają dostęp praktycznie z dowolnego miejsca. Z tych dwóch powyższych tylko jedno zapewnia w miarę stałą szybkość transmisji. Tym łączem jest łącze telefoniczne. Wskazując na łącze telefoniczne, rodzi się następny problem. Mianowicie, przesłanie np. 512 KB danych przez telefon przy użyciu modemu na 33.6 kb/s trwałoby około (zakładając transfer na poziomie 3 kB/s) 3 minut. Przy tak długiej transmisji problematyczna byłaby kontrola i sterowanie na odległość większością aplikacji. Skoro nie można zwiększyć szybkości transmisji, to należy ograniczyć ilość danych na drodze kompresji [6] [7] [8].

Ilość danych niezbędnych do opisu obrazu jest dość duża i wzrasta szybko wraz z polepszaniem się rozdzielczości, oraz poprawą odwzorowania barwy każdego piksela. Stosując różne metody kompresji (w zasadzie usuwające część nadmiarową informacji), można zredukować ilość informacji do przesłania poprzez łącze transmisyjne. Stosowane metody kompresji obrazów można podzielić na metody bezstratne oraz metody stratne.

Metody kodowania bezstratnego (w przypadku grafiki) charakteryzują się niewielkim stopniem kompresji ok. 40 - 50% [6]. Do metod tych zaliczyć można: kodowanie ciągów symboli, kodowanie drzewiaste, kodowanie konturowe, metodę statystyczną, kodowanie słownikowe (np. algorytm Lempela-Ziva, algorytm Lempela-Ziva-Wellcha) i inne. Metody kompresji bezstratnej stosowane są wszędzie tam, gdzie pożądana jest prostota algorytmów oraz niepożądane wprowadzanie do obrazu zakłóceń.

Znacznie lepsze wyniki kompresji można osiągnąć, jeżeli dopuszcza się możliwość wprowadzenia pewnych modyfikacji w kodowanym obrazie (w zasadzie utraty części informacji). Oko ludzkie ze względu na swoją niedoskonałość nie jest w stanie zarejestrować niewielkich zmian, a w szczególności niedostrzegalne są dla ludzkiego oka drobne zmiany wartości niektórych pikseli. Powyższe spostrzeżenie stało się podstawą algorytmów stratnych kompresji obrazów. Do metod kodowania modyfikujących obraz można zaliczyć: kodowanie różnic, kodowanie blokowe, zmniejszanie liczby barw, przekształcenia afiniczne, JPEG (wykorzystuje się transformatę kosinusową), MPEG (kompresja sekwencji obrazów) i inne. Głównym ograniczeniem stosowania metod stratnych jest ich złożoność obliczeniowa, a w szczególności często proces kompresji trwa znacznie dłużej od procesu dekompresji.

Decydując o wyborze jednej z metod kompresji, należy mieć na uwadze fakt, że zdalna kontrola jest tylko dodatkową aplikacją działającą w tle i nie może zbytnio obciążać procesora, gdyż głównym zadaniem jest wykonywanie aplikacji sterująco-monitorującej. Stąd algorytm kompresji obrazu graficznego nie powinien być zbyt skomplikowany, a w żadnym przypadku nie powinien spowalniać lub wstrzymywać działania aplikacji pierwszoplanowej. Niestety, prostsze algorytmy (a takimi są kompresje bezstratne) nie gwarantują odpowiedniego stopnia kompresji [6].

3. Przykłady rozwiązań aplikacji umożliwiających zdalną kontrolę

Jedną z uniwersalnych aplikacji pozwalającą na sterowanie i kontrolowanie pracą komputera klasy PC (a tym samym umożliwiającą uruchamianie i sterowanie innymi aplikacjami na odległość) poprzez wiele rodzajów łączy jest produkt firmy Symantec o nazwie NORTON pcANYWHERE [2]. Podstawową zasadą działania tego oprogramowania jest przesyłanie ekranu komputera kontrolowanego do komputera odległego poprzez wybrane łącze transmisyjne, oraz przesyłanie w drugą stronę kodów z klawiatury komputera odległego. Oprogramowanie umożliwia pracę z dowolnymi aplikacjami zarówno w trybie tekstowym, jak i graficznym. Przed wysłaniem zawartość pamięci ekranu poddana jest zabiegowi kompresji metodą kodowania ciągów identycznych wartości (dla ekranów w trybie tekstowym). Przesyłanie ekranu tekstowego nie stwarza większego problemu nawet poprzez

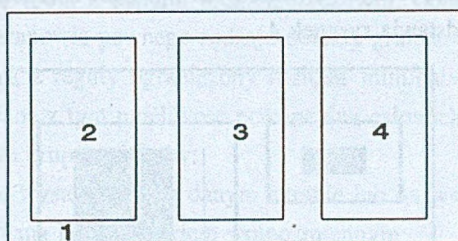
modem, ponieważ ilość danych jest stosunkowo niewielka (4 KB). Natomiast przesyłanie obrazów graficznych jest już pewnym problemem z uwagi na ilość danych i małą sprawność zastosowanej metody kompresji (ok. 30 - 40%). Przykładowe transmisje ekranów graficznych zebrane zostały w tabeli 3.

Tabela 3

Testy transmisji pamięci obrazu

Tryb	Rozmiar pamięci ekranu	Ilość przesłanych danych pomiędzy komputerem typu host a komputerem odległym dla pojedynczego testowego ekranu
CGA 4 kolory	16 KB	12,3 KB
EGA 16 kolorów	32 KB	20 KB
VGA 256 kolorów	64 KB	38 KB

W tabeli 3 nie uwzględnione są dodatkowe dane wynikające ze sterowania i przesyłania potwierzeń. Na rysunku 3 przedstawiono wygląd jednego z testowych ekranów graficznych.



- 1- tło ekranu czarne
- 2- obszar o kolorze żółtym
- 3- obszar o kolorze czerwonym
- 4- obszar o kolorze zielonym

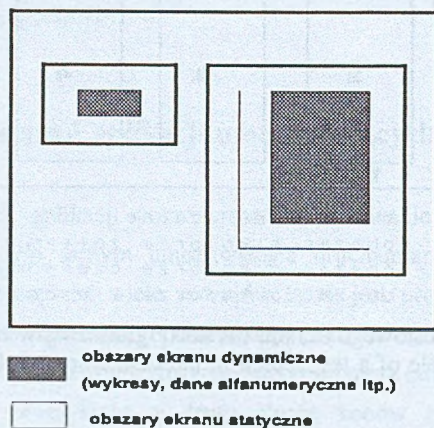
Rys. 3. Przykład testowego ekranu dla karty graficznej w trybie CGA 4 kolory
Fig. 3. The example of a test screen in the case when works in the CGA 4 colors mode

Jak widać, zastosowana metoda kompresji nie jest zbyt efektywna dla prostych ekranów graficznych. Wyświetlanie odbywa się sukcesywnie wraz z napływającymi danymi, linia po linii. Ujemnym efektem działania programu jest chwilowe wstrzymywanie wykonywania aplikacji na kontrolowanym komputerze w czasie obróbki danych ekranu tym dłużej, im bardziej skomplikowany jest obraz, oraz im większa jest jego rozdzielczość.

Drugim przykładem podobnie działającej aplikacji jest asynchroniczna konsola sieciowego systemu operacyjnego Novell NetWare, służąca do administrowania serwerem. Tutaj jednak transmitowane są tylko ekrany w trybie znakowym, przy czym w przeciwieństwie do poprzedniego programu wyświetlanie obrazu na komputerze odległym rozpoczyna się dopiero po przesłaniu wszystkich informacji związanych z danym ekranem.

4. Koncepcja skrócenia czasu transmisji ekranu poprzez zmianę metody kompresji

Istnieje dość liczna grupa prostych aplikacji opracowywanych przez małe firmy lub pojedynczych programistów, gdzie ma miejsce graficzna wizualizacja zachodzących procesów sterowania. Można zauważyć, że ekrany graficzne często tworzone są dla standardowego trybu VGA o rozdzielczości 640x480 przy 16 kolorach, co daje 156 kB pamięci obrazu. Dodatkowo ekran (lub kilka ekranów) wykazuje pewną prostotę (obszary ekranu składają się z regularnych figur geometrycznych o jednolitych wyraźnie zaznaczonych powierzchniach). Przykładowy ekran przedstawia rysunek 4.



Rys. 4. Przykładowy ekran graficzny

Fig. 4. An example of the graphical screen

Zastosowanie prostych algorytmów kompresji dla takich ekranów nie zmniejsza w oczekiwany sposób rozmiaru danych do przesłania (przy 40% kompresji dane miałyby rozmiar ok. 93 kB) i przy zakładanej szybkości transmisji np.: 3 kB/s (jeżeli tym łączem jest analogowe łącze telefoniczne, transmisja trwałaby ok 31 sekund). Przy 31 sekundach potrzebnych do transmisji pojedynczego ekranu sterowanie pracą komputera na odległość może być niemożliwe, jeśli zmiany treści ekranu następowałyby w czasie krótszym niż czas potrzebny na transmisję danego ekranu. Aby nie zgubić żadnej zmiany w przesyłanych ekranach, zmiany elementów obrazu powinny następować nie częściej niż co trzydzieści parę sekund, lub musiałoby następować wstrzymywanie wykonywania aplikacji na kontrolowanym komputerze do czasu zakończenia transmisji bieżącego ekranu. Czas oczekiwania na ekran można skrócić poprzez zastosowanie bardziej efektywnej metody kompresji. Nowa metoda kompresji nie może jednak w istotny sposób zwiększać obciążenia procesora stacji kontrolowanej. Wobec powyższego, jako metodę kompresji proponuję zmodyfikowaną metodę kompresji opartą na metodzie słownikowej mieszanej statycznie-adaptacyjnej zorientowanej obiektowo. Elementami słownika nie powinien być piksel lub grupa pikseli, lecz coś znacznie większego, np: charakterystyczne obiekty analizowanego obrazu. Ekran graficzny, tworzone dla potrzeb wizualizacji procesów przemysłowych, wyróżniają się spośród innych typów ekranów graficznych pewnymi charakterystycznymi cechami, jak:

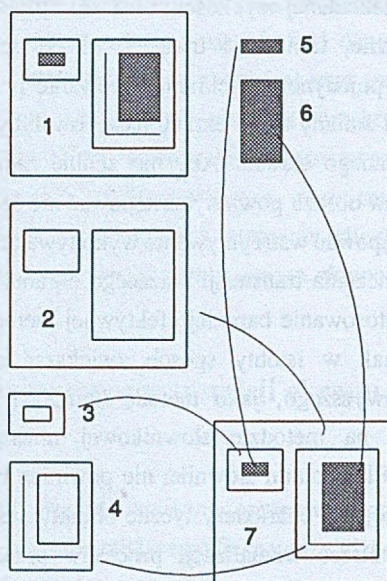
- stosunkowo duża prostota i przejrzystość (duży stopień uporządkowania obrazu);
- ograniczona liczba wyraźnie zaznaczonych obszarów niosących w swojej treści jakąś konkretną informację (pola z danymi alfanumerycznymi, pola wykonawcze, schemat instalacji itp., które stanowią pewnego rodzaju obiekty graficzne);
- obiekty ekranu mają z reguły ograniczony rozmiar minimalny z uwagi na charakter zastosowania i związaną z tym możliwość percepcyjną człowieka.

Można wyróżnić dwie grupy obiektów:

- a) statyczne, stale występujące na danym ekranie lub na wszystkich ekranach, oraz takie, których zmiana jest zjawiskiem wolnoźmiennym;
- b) dynamiczne, w których zachodzi dynamiczna duża zmienność wypełnienia wewnętrznego obszaru obiektu (np. alfanumeryczna prezentacja jakiegoś szybkozmiennego parametru).

Dla przykładowego ekranu z rysunku 4 obiektami statycznymi byłyby wszystkie obszary prostokątne tego ekranu. Zastępując te obiekty indeksami słownika, można by opisać taki ekran niewielką ilością bajtów. Pozostałe elementy zmienne, obrazujące stan procesu (w podanym przykładzie wykresy, wartość parametrów - czyli obiekty dynamiczne) przesyłać można jako nałożenie w ściśle zdefiniowany obszar ekranu, a dokładnie w ściśle określony obszar już zdefiniowanego obiektu. Algorytm kompresji dla danej aplikacji poszukiwałby w obrazie tylko znanych elementów słownika. Dla danej aplikacji słownik byłby częściowo

niezmienny, tworzony raz przed uruchomieniem zdalnej kontroli. Przykład takiego działania przedstawia rysunek 5.



Rys. 5. Przykład działania kompresji opartej na metodzie słownika obiektów
Fig. 5. The compression, based on the objects dictionary compression method

Na rysunku 5 przez 1 oznaczyłem analizowany ekran, a przez 7 ekran otrzymany na stacji odległej. Zastosowany algorytm kompresji wyszukuje w analizowanym ekranie (1) znane i wcześniej zdefiniowane obiekty statyczne (2,3 i 4) oraz dynamiczne (5 i 6). Jako dane transmitowane są do stacji odległej ciągi kodowe odpowiadające elementom statycznym obrazu, oraz ciągi kodowe odpowiadające elementom dynamicznym analizowanego ekranu. Dla przesłania informacji o elemencie statycznym wystarcza przesłanie ciągu kodowego związanego z danym elementem, natomiast dla elementu dynamicznego niezbędne jest przesłanie, oprócz ciągu kodowego identyfikującego wystąpienie danego elementu, również danych związanych z treścią ekranu tego elementu. Elementami statycznymi słownika w omawianym przykładzie są elementy oznaczone przez 2, 3 i 4. Elementy zmienne słownika oznaczone są przez 5 i 6.

Dla potrzeb prezentowanej metody można przyjąć następujący format danych:

- bajt sterujący (bieżący ekran, retransmisja, cały ekran obiektem dynamicznym);
- dwubajtowy indeks każdego obiektu statycznego (typ i położenie);

- pięciobajtowy indeks każdego obiektu dynamicznego (typ, położenie i długość danych reprezentujących dany obiekt dynamiczny) z dołączonymi danymi reprezentującymi obiekt dynamiczny.

Jakie korzyści przynosi zaproponowana metoda kompresji? Przede wszystkim radykalne zmniejszenie ilości transmitowanych danych, będących wynikiem znacznego wzrostu stopnia kompresji. Jeżeli w obrazie można by było wyróżnić kilka lub kilkanaście obiektów, będących elementami statycznymi słownika, jednocześnie obiekty te stanowiłyby zdecydowaną większość danego ekranu, to do ich zakodowania wystarczyłoby kilkadziesiąt bajtów. Do przetransmitowania ekranu do stacji odległej z rysunku 3 wystarczyłoby kilkanaście bajtów, wraz z bajtami sterującymi (dokładnie 9 bajtów), zamiast 12 kB. Natomiast dla przykładowego ekranu z rysunku 5 ilość danych transmitowanych łącznie wyniosłaby 17 bajtów danych związanych z indeksami obiektów oraz dane reprezentujące obiekty dynamiczne (ok. 10% danych reprezentujących cały ekran).

Obszar zastosowania powyższej metody kompresji ograniczałby się do aplikacji, w których ekrany graficzne są stosunkowo proste, złożone z obiektów (obszarów) o regularnych kształtach, oraz szybkość zmian elementów ekranu nie byłaby zbyt szybka (w stosunku do czasu potrzebnego na transmisję pojedynczego ekranu).

Rozwiązując powyższe, należy rozstrzygnąć między innymi następujące problemy:

- W jaki sposób tworzyć słownik dla danego zastosowania? Czy tworzyć go poprzez dedykowaną aplikację (zaletą tego rozwiązania byłaby większa uniwersalność, ale wadą niewątpliwie większa czasochłonność)? Czy opracować zasady, zgodnie z którymi, programista piszący oprogramowanie sterująco-monitorujące tworzyłby jednocześnie bibliotekę obiektów obrazów graficznych swojej aplikacji, czyli tym samym słownik dla algorytmu kompresji (zawęzałoby to zakres zastosowań, ale zaletą byłoby skrócenie czasu związanego z przygotowaniem aplikacji zdalnej kontroli do pracy i sprowadzałoby się do - poprzedzającej właściwą pracę - dodatkowej transmisji związanej z biblioteką obiektów)?

- Czy transmisję ograniczyć tylko do kombinacji kodowych identyfikujących dane elementy ekranu (ujemną stroną takiego rozwiązania byłby dłuższy słownik w przypadku wielu różnych ekranów z występującym takim samym elementem, ale inaczej umiejscowionym, zaletą natomiast większy stopień kompresji)? Czy transmitować kombinacje kodowe identyfikujące obiekt wraz z opisem położenia danego elementu (słownik niewątpliwie krótszy dla przypadku analogicznego jak poprzedni, ale stopień kompresji mniejszy, oraz bardziej skomplikowany algorytm analizujący dany ekran)?

5. Wnioski

Jeżeli z przyczyn obiektywnych nie można zwiększyć szybkości transmisji, to należy zastosować bardziej efektywną metodę kompresji. Zaproponowana metoda zapewnia redukcję danych na znacznie większym poziomie niż dotychczas stosowane metody, co pokazano w przykładach w punkcie 4 artykułu. Stopień kompresji zależy tutaj od procentowego udziału obiektów zdefiniowanych jako dynamiczne w treści całego ekranu graficznego. W skrajnym przypadku, tzn. przy braku obiektów dynamicznych, redukcja danych przekracza 90% i nie zależy od rozdzielczości i liczby kolorów takiego ekranu. Z drugiej strony, w przypadku kiedy cały ekran będziemy traktować jako obiekt dynamiczny, uzyskamy redukcję danych na takim poziomie, na jaki pozwolą nam dotychczasowe metody kompresji.

LITERATURA

1. Sikora H., Steinparz F.X.: *Computer & Kommunikation*, Hanser 1988.
2. *User's Guide: The NORTON pcANYWHERE*, Symantec Corporation 1995-98.
3. Ziółkowski K., Twardoń A.: *Zdalny dostęp i usługi komunikacyjne w systemach Windows 95 i Windows NT*. ZN Pol. Śl., Seria Informatyka z.30 (s. 29-43), Gliwice 1996.
4. Jarża R.: *Sterowniki EGA, VGA, SuperVGA przewodnik dla programistów*, Wrocław 1993.
5. Bulhak L.: *DOS 5.00/6.22 od środka*, Warszawa 1992.
6. Praca zbiorowa pod redakcją Zabrodzkiego J.: *Grafika komputerowa metody i narzędzia*, WNT, Warszawa 1994.
7. Śluzek A.: *Komputerowa analiza obrazów*, WPW, Warszawa 1991.
8. Nelson M.: *The data compression book*, Prentice-Hall 1991.
9. *Windows NT: System Guide*, Microsoft Corporation 1995.
10. Piasecki J.: *Zdalny dostęp i zdalna kontrola w lokalnych sieciach komputerowych*. ZN Pol. Śl., Seria Informatyka z.33 (s. 165-173), Gliwice 1997.
11. Tanenbaum A.S.: *Sieci komputerowe*, WNT, Warszawa 1988.
12. Praca zbiorowa pod red. Golańskiego R.: *Wybrane systemy i układy scalone w telekomunikacji cyfrowej*, AGH, Kraków 1995.

Recenzent: Dr inż. Ryszard Winiarczyk

Abstract

In the paper some problems connected with the transmission of the display memory in remote control applications were presented. The comfort and rate of remote control depend on:

- transmission rate (in this case the selection of a medium is important);
- compression of the graphical screens.

The compression of the graphical screens seems to be more important because it leads up to data reduction before transfer. The commonly used methods of lossless compression are not satisfying (40 % data reduction is not enough). The static-adaptive, object orientated leksical method presented in the paper allows to data reduction on the 90 % level. Such a compression is enough for remote process control even with telephone medium.